Geometric Message Passing to utilize local and non-local information in 3D Graph Networks

Svødstein, Jákup Group mi908e20, Master student at the Department of Computer Science, Aalborg University

***Abstract*—To revolutionize computational chemistry within simulating and analysing molecular systems. A principled framework that captures the relative 3D information and long-range interactions is needed.**

In this work, we propose a generic framework to capture these interactions, known as the deterministic point graph network (DPGN). It provides a unified interface to interact with 3D graphs on different levels of bonds, angles, torsional and long-range effects. We then leverage this framework to add multiple structural improvements to propose the geometric message passing scheme (GMP) to realize DPGN. We demonstrate the benefit of the proposed changes in ablation studies. Finally, we validate this model by presenting results beating the predecessor MPNet on six properties on the QM9 data set, whereas one of them is state-of-art. Additionally, we demonstrate the models' ability within the field of molecular dynamics, where it beats state of the art on 50% of the targets on in MD17 data set.

I. Introduction

Computational quantum mechanical modelling methods are of interest within physics, chemistry, and earth sciences to investigate various quantum chemical properties. Computing the close form solution of the many-body Schrödinger equation for quantum chemical properties are prohibitively expensive for all but the simplest molecules.

To get insight into systems with higher complexities, researchers have been able to approximate the chemical properties with classical algorithms such as Density Functional Theory (DFT) [1] by sacrificing prediction accuracy to reduce the computational cost.

Recently, researchers have been developing machine learning methods that are several orders of magnitude faster than DFT without sacrificing prediction accuracy.

Among these machine learning approaches, graph neural network (GNN) based methods have gained most research attention due to their ability to model complex interactions between atoms without being restricted to predicting equilibrium structures [2, 3].

The early era GNNs developed for learning on graph data were later unified into a general graph network framework (GN) [4] by using a message-passing architecture. GN is developed for regular 2D graphs, rather than 3D graphs. Therefore, its message-passing update function only depends on the previous atom embeddings and the pairwise distances to neighbouring atoms E_{bonds}.

An architecture with this restriction has to rely on complex higher-order interactions to capture spatial information like angles between neighbours E_{angles}, dihedral angles between the plane of bonds $E_{dihedral}$ or non-local, non-covalent interaction between atom pairs $E_{non-local}$ [2, 3].

Whereas a message-passing scheme for 3D graph structures, can derive various kinds of relative 3D information, meanwhile ensuring their invariance to rotation, translation and inversion [5]–[7].

One way to ensure these invariances is by using inductive biases from the insights given by the domain of interest.

Within quantum chemistry, this has led to GNNs and message passing schemes which are roto-translational and permutation equivariant, also known as equivariant graph neural networks (EGNN) [8]–[16].

3DGN extended GN's architecture, to give it the ability to work with relative 3D information [17].

In this work, we propose the deterministic point-cloud graph network (DPGN). It is a generic framework for 3D graphs for deterministic point cloud systems, which extends MPNet to fully utilize the 3D graph structures and is integrated into the framework set by 3DGN. The goal is to contribute to the groundwork for developing a complete pipeline for simulating and analysing molecular systems, by incorporating the energy from both the local and non-local environment meanwhile staying within EGNNs constraints [18].

$$E = E_{local} + E_{non-local} \tag{1}$$

where $E_{local} = E_{bonds} + E_{angles} + E_{dihedral}$.

We note that it is not ideal to use a Cartesian coordinate representation as inputs for 3D graphs. The tight coupling between the reference frame and the physical system it presents makes it not invariant to roto-translations of input graphs.

To preserve the euclidean symmetries as needed for an EGNN, we use a tensor representation that is equivariant to the rotation group $SO(3)$ and incorporates inductive biases based on the rotation, translation and inversion symmetries.

We propose a novel message passing scheme known as the geometric message passing scheme (GMP), for realizing the DPGN framework. The relative positional information is encoded by taking their irreducible representation of $SO(3)$ and using it as a directional embedding. To keep the information roto-translational and permutation equivariant, its encoding is jointly represented by the use of spherical Bessel functions and spherical harmonics functions [9].

Additionally, for GMP to learn both atomic forces and molecular properties and be preserving the relative positional information through the message passing convolutions, it is constrained to be twice continuously differentiable [9].

We apply the GMP to the data sets QM9, COLL and MD17, where physical representations are essential to demonstrate the performance of the new method. Ablation study reveals the contributions of the various 3D positional information seen in Eq.(1)

II. Related work

A. Graph neural network

Graph Neural Networks (GNNs) was first proposed in the late 90s [19, 20] and 00s [21, 22] and is being used as a general framework to model complex graph-structured data using neural networks [23]–[25].

Notable methods include GCN [25], GIN [26], GAT [27], GGNN [28], NLNN [29]. Currently, the message passing networks (MPNNs) is the most general architecture for realising a GNN [2]. Within this method, the nodes state is updated by aggregating messages flowing from its direct neighbours. The other variances can mostly be realised by changing the way each node aggregates the representations of its neighbours with its representation. The GN framework [4] is the latest unification within generalizing GNNs. It does so by generalizing and extending various graph neural network architectures by reframing the computations to their constituent parts and performing computations over the graph's nodes, edges, and system-level properties by global attributes.

GN is incapable to use a relative 3D positional data structure, it has to capture the effect bond lengths, angles between bonds, dihedral angles or long-range non-covalent forces. This implicit adaptation must learn these effects through aggregating the messages from its neighbourhood.

3DGN extended GN to account for 3D graph networks [17]. It does so by encoding relative 3D spatial information into the framework. We are reframing the revised version of MPNet into the 3DGN architecture while taking into account inductive biases like symmetry groups and relative 3D spatial information and long-range forces.

B. Machine learning for molecules

The initial works using data-driven methods within computational chemistry had an emphasis on reducing the set of reference calculations when developing inter-atomic potentials or predicting chemical properties. This was done by using descriptor-based approaches (e.g., Coulomb matrix, bag of bonds), shallow neural networks with Gaussian processes designed to keep the model invariant concerning roto-translation and permutation to develop inter-atomic potentials or predict chemical properties [30, 31].

Recently, rotational invariant graph neural networks (GNN-RI) have emerged as an architecture for developing inter-atomic potentials while eliminating the need for hand-crafted descriptors. Instead, it can learn invariant features of geometric data by the representational form of the molecules on the graphs. [3, 9, 32, 33] GNN-RIs typically connects every atom to all other atoms, if their inter-atomic distance is closer than a variable cutoff distance r_c. This restriction localizes the problem space, to reduce the computational complexity and prevent over-fitting. Within computational chemistry, most of the effects can be captured through aggregating information of the neighbours instead of creating a complete graph.

Every node is associated with a scalar feature $h_i \in \mathbb{R}^h$ which iteratively gets updated by aggregating the messages of the neighboring atoms r_{ij} and their feature vectors $h_j \in \mathbb{R}^h$.

One unfortunate consequence of having a variable cutoff distance is loss of expressivity since it no longer can distinguish certain molecular structures. A way to prevent the loss of expressivity is to enable GNN's to utilise angular information, DimeNet expands the message passing scheme by adding angular information and three-body terms to the pairwise distances [9], HMGNN incorporates many-body interactions by using heterogeneous nodes and edges [1], MPNet tries to capture non-covalent interactions by using radial basis functions for two layers with different cutoff distances, which embed inter-atomic distances, which get unified through a multi-layered perception [34]. MXMNet tries to capture non-covalent interactions through multiplexing two layers with different cutoff distances. We use MPNets approach to capture the non-covalent forces.

C. Equivariant neural networks

Equivariance and invariance have become one of the core principles within physics-informed neural networks, for which induced biases for various symmetries to reduce redundancy in the space of functions the models need to take into consideration. Equivariant models for the $SO(3)$ were first explored in the context of spherical convolutions, by leveraging the representational power of the $SO(3)$ group [8, 15, 35]. Afterwards, they were used in the context of 3D point clouds and property predictions of molecules [8, 12, 36]–[39].

Importantly, Dym & Maron [40] proved the universality of models for a joined group of translations, rotations, and permutations. Which, apart from reflections, is the group relevant for molecules.

Finally, when working with the special case of rotationally invariant predictions, Klicpera et al. [41]. has proven that any rotationally invariant function can be approximated on S^2 without losing the universality property.

Thus, a directional representation is adequate and will be used in this work, since it is possible to achieve the best results without relying on a more expressive representation while working within these constraints.

III. Notation and Definitions

Since DPGN is built upon MPNet, we use the same notation and definitions [34] We denote matrices by bold upper-case letter (e.g., $\boldsymbol{W}$) and vectors by lower case letters (e.g., $\boldsymbol{x}$). Entries in matrices and vectors are represented with subscripts (e.g., x_{ij}). for variables at the t-th message passing layer (e.g., $\boldsymbol{h}^{(t)}$).

We denote a molecular graph as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{N} = |\mathcal{V}|$ nodes (*atoms*) and $\mathcal{M} = |\mathcal{E}|$ edges.

Node $\mathcal{N}_i$ is connected to the nearest neighbour nodes in the *molecular graph*, if $\mathcal{N}_i = \{i|d(i,j) < c\}$, where $d(i,j)$ is the Euclidean distance between node i and j, where j is within the cutoff threshold $c > 0$. The set $\mathcal{E}$ stores the whole geometric structure of the molecule, since every edge is associated with distances, as $\forall i,j \in \mathcal{E} : i \neq j \rightarrow d(i,j) \leq c$.

IV. A Framework for deterministic 3D Graphs

Within the natural sciences, deterministic point cloud data points can naturally be represented as 3D graphs. For instance, molecular representation learning is an essential task for fields like biochemistry and material science [42, 43].

A common representational form when modelling molecules as graphs is to connect every atom/node with an edge to all other atoms/nodes if their interatomic distance is closer than the cutoff distance c. Whereas a molecule is uniquely defined by its atomic numbers $\boldsymbol{z} = \{z_1, ..., z_N\}$ and its Cartesian coordinates $\boldsymbol{X} = \{x_1, .., x_n\}$. These nodes and edges can contain attributes, to hold relative spatial information, atom types or other auxiliary information $\Theta \in \mathbb{R}^n$, where n is the number of attributes. In this work, the focus is on scalar regression targets $t \in \mathbb{R}$, without the use of auxiliary features, since the goal is to stay with a generic framework with the ability to cover various domains without modifications to the architecture.

A. Symmetries and invariances

Every physical system which is continuous and smooth with a differentiable action has a correspondent conservation law. When developing a neural network to model physical systems by using pairwise distances, yields a framework with conservation of momentum, energy and angular momentum.

In this work the symmetries of interest are: *Roto-translational symmetry* which follows from the conservation of momentum, energy and angular momentum.

Parity symmetry as a consequence of the decoupling of the representational framework and the physical system.

Permutation invariance to have the ability to indistinctly aggregate the messages from the neighbour particles.

Any neural network can learn these symmetries implicitly [44]. However, doing so most often leads to a sub-optimal model with duplicate weights, which consequently leads to longer training times.

To circumvent this, it is possible to constrain the models by explicitly incorporating these symmetries by using inductive biases. The invariant representational form used in this paper is 3D-geometric tensors, whereas the features are encoded into the filters of a spherical harmonic function by using the tensor rank as the angular frequency with a learned radial basis function.

B. Definition of a 3D graph

Within the DPGN framework, a 3D graph is defined as a 4-tuple $\mathcal{G} = (\boldsymbol{u}, \boldsymbol{V}, \boldsymbol{E}, P)$. It consists of the global feature vector $\boldsymbol{u} \in \mathbb{R}^k$; for example, $\boldsymbol{u}$ might represent the electro-magnetic field.

The $\boldsymbol{V}$ is the set of node features $v_i, \forall i \in \{1, ..., N^v\}$ (of cardinality N^v) where $v_i \in \mathbb{R}^d$ is the feature vector for node i. For example, $\boldsymbol{V}$ might represent each atom, with various physical properties.

The $\boldsymbol{E}$ is the set of edges $(e_k, r_k, s_k), \forall k \in \{1, ..., N^e\}$ (of cardinality N^e) where $e_k \in \mathbb{R}^d$ is the feature vector,

r_k is the index of the receiver node, and s_k is the index of the sender node for the edge i. For example, $\boldsymbol{E}$ might represent the bonding strength between two atoms.

The P is the set of coordinates x_i, $\forall i \in 1, ..., N^p$, (of cardinality N^p).

Lastly, we define $E_i = (e_k, r_k, s_k), r_k = i \ \forall k \in \{1, ..., m\}$ as the set of edges pointing to node i, and $\mathcal{N}_i = j | d(i,j) < c$ as the set of indices of the incoming nodes of node i, where $d(i,j) = |x_i - x_j|^2$ is the euclidean distance between node i and j.

C. Internal structure of a DPGN block

The information flow through a DPGN block consists of transformations through a set of three update functions Ψ and a set of six aggregation functions ρ, the computational steps can be seen in Fig. 1.

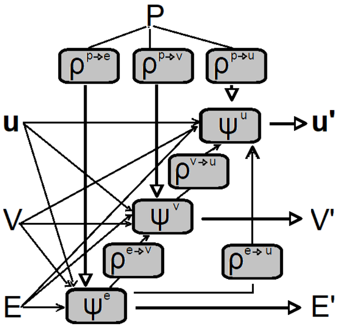

Fig. 1. Illustration of the information flow through a DPGN Block

The Ψ functions maps the information for the corresponding geometry, whereas the ρ functions take a set as input and reduce it to a single element representing the aggregated information, as described in Eq. (2).

$$\begin{aligned} e_k^{'} &= \Psi^e(e_k, v_{r_k}, v_{s_k}, \boldsymbol{E}_{s_k}, u, \rho^{p\rightarrow e}(\{r_h\}_{h=r_k \cup s_k \cup \mathcal{N}_{s_k}})) \\ v_i^{'} &= \Psi^v(v_i, \rho^{e\rightarrow v}(E_i), \boldsymbol{u}, \rho^{p\rightarrow v}(\{r_h\}_{h=i \cup \mathcal{N}_i})) \\ u^{'} &= \Psi^v(\rho^{e\rightarrow u}(E^{'}), \rho^{u\rightarrow v}(V^{'}), \boldsymbol{u}, \rho^{p\rightarrow u}(\{r_h\}_{h=i:n})) \end{aligned} \tag{2}$$

where

$$\boldsymbol{E_i^{'}} = (e_k, r_k, s_k), r_k = i, \forall k \in \{1, ..., N^e\},$$

$$\boldsymbol{V'} = v_i^{'}, \forall i \in \{1, ..., N^v\},$$

and

$$\boldsymbol{E}^{'} = \bigcup_i \boldsymbol{E_i^{'}} = \{(e_k^{'}, r_k, s_k)\}, \forall k \in \{1, ..., N^e\}.$$

D. Computational steps within the unified message passing scheme

When a graph, $\mathcal{G}$ is provided as input to a DPGN block, the computations proceed from edges to nodes, to the global level.

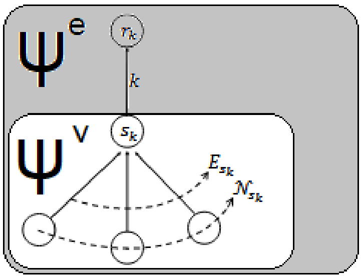

Fig. 2. Illustration of functions Ψ^e and Ψ^v

The Ψ^e function maps the embedding of every edge k, $e_k \rightarrow \boldsymbol{e}_k^{'}$. Its input parameters are the old edge vector e_k, the target node v_{t_k}, the sender node v_{s_k}, the global feature vector $\boldsymbol{u}$, the set of edges $\boldsymbol{E}_{s_k}$, and the positional information on all the nodes connected by the edge k and edges in E_{s_k}, with the index set as $t_k \cup s_k \cup \mathcal{N}_{s_k}$. Ψ^e with the indices is provided in Fig. 2

the $\rho^{p\rightarrow e}$ aggregates the embeddings from these nodes to update edge k.

The Ψ^v function is mapped across all the nodes to compute per-node updates as $v_i \rightarrow v_i^{'}$. The input parameters are the old node vector v_i, the set of edges $\boldsymbol{E}_i$ pointing to node i, the global feature vector $\boldsymbol{u}$, and the positional information on all the nodes connected by the edge k with the index set as $i \cup \mathcal{N}_i$. As can be seen in Fig. 2, the function Ψ^v is encapsulated in the white section. By looking at the node with index i instead of s_k.

The $\rho^{e\rightarrow v}$ and $\rho^{p\rightarrow v}$ aggregate the embeddings of the input edge features to input nodes. The Ψ^u is the global update function $\boldsymbol{u} \rightarrow \boldsymbol{u}^{'}$ applied to the whole graph feature once, while the aggregate functions $\rho^{p\rightarrow e}$, $\rho^{p\rightarrow v}$ and $\rho^{p\rightarrow e}$ aggregate the information from edges, nodes and positional information about the node features. It is important to note, ρ functions must be invariant to permutations of their inputs and have the ability to take a variable amount of arguments, like the element-wise sum. Whereas the Ψ functions are not constrained and can be implemented in different ways.

V. Geometric message-passing Neural Networks

GNN's for molecules when working with a point cloud with n points (atoms), the directional information is typically incorporated in one of two ways:

Via $SO(3)$ representations, which is the double cover from the four-dimensional S^3 sphere, or by using a directional representation associated with the three-dimensional S^2. Thus, the directional representation has one degree of freedom less than a $SO(3)$ representation, making it cheaper at the price of expressivity.

Since we solely are interested in the special case of rotationally invariant predictions, the directional embedding can approximate any rotationally invariant function on S^2, which is exactly what we need. Therefore, using the directional embeddings with one degree of freedom less is adequate, and is what we use we use in this work.

A. Geometric message-passing

1) Geometric Representation: To get a directional representation, the positional information contained in P is incorporated into the aggregate functions $\rho^{p\to e}$, $\rho^{p\to v}$, and $\rho^{p\to v}$ which maps the positional information on the different geometries.

Note: the positional information in P is not roto-translationally invariant, since its positional information is in Cartesian coordinates. To resolve this issue, P is converted into a representation in the spherical coordinate system (SCS).

In SCS, the location of any point is specified by a 3-tuple (d, θ, ϕ), where d is the radial distance, θ is the polar angle, and ϕ is the azimuthal angle, respectively.

Within this representation, it is possible to treat any node i as the origin of a local representation. Thus, the relative location of each neighbouring node of i is specified by its tuple(d, θ, ϕ), denoting the edge length, the angle between edges, and the torsional angle.

Consequently, a directional representational graph structure in SCS is invariant to translation and rotation of the input graph, which is exactly what we need.

However, the obtained SCS representation cannot be used as the direct input to a neural network, due to its lack of being a meaningful physical representation. This can be circumvented by embedding the relative directional information into a geometric representation $(d, \theta, \phi) \to \psi(d, \theta, \phi)$.

For this, we use the Spherical Fourier-Bessel basis introduced by Klicpera et al. [9]. This basis has its grounding in Schrödingers time-independent wave function in SCS from quantum mechanics and is defined as following:

$$\psi(d, \theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{m=l} a_{lm} jl(kd) Y_l^m(\theta, \phi) \tag{3}$$

where Y_l^m is a spherical harmonic function of degree m and order l, a_{lm} is a set of coefficients regarding l and m, and j_l is a spherical Bessel function, of order l. We use the same orthogonal basis for j_l with the boundary condition $k = \frac{z_{ln}}{c_r}$, where z_{ln} is a the n-th root of the l-order Bessel function, c_r denotes the cutoff value, $l \in [0, ..., N_{SHBF} - 1]$, and $n \in [1, ..., N_{SRBF}]$, where N_{SHBF} and N_{SRBF} denote the highest orders for the spherical harmonic and spherical Bessel functions, they are given as hyper-parameters in the experimental setting.

A geometric representation which preserves all available geometric information in 3D is obtained by splitting this basis into the radial, circular and spherical part between all atom pairs within the distance $x_{ca} \leq c_r$, as following:

$$\tilde{e}_{RBF,n}(d) = \sqrt{\frac{a}{c_r}} \frac{\sin(\frac{n\pi}{c_r} d)}{d} \tag{4}$$

$$\tilde{e}_{CBF,ln}(d, \theta) = \sqrt{\frac{2}{c_{int}^3 j_{l+1}^2(z_{ln})}} jl(\frac{z_l n}{c_{int}} d) Y_l^0(\theta) \tag{5}$$

$$\tilde{e}_{SBF,lmn}(d, \theta, \phi) = \sqrt{\frac{2}{c_r^3 j_{l+1}^2(z_{ln})}} jl(\frac{z_l n}{c_r} d) Y_l^m(\theta) \tag{6}$$

The difference between the global RBF and the local RBF is the value of c_r in Eq. (5), where $c_{global} \geq c_{local}$.

Geometric message-passing Scheme. We now develop the geometric message-passing scheme (GMP) for realizing the DPGN framework. To obtain the geometric representation, let v_{sk} be a sender node, which serves as the node at the origin of a local SCS. v_{sk} sends a message e_k to the target node v_{tk}. The message e_k is updated based on $\boldsymbol{E}_{s_k}$ from $\mathcal{N}_{s_k}$.

Lets define $q \in \boldsymbol{E}_i$, as a tuple (d, θ, ϕ), where d is the distance between them and θ denotes the direction to update the message e_k.

Lets denote $o \in \mathcal{N}_{s_k}$ as the reference node needed to compute ϕ.

Then the torsional angle ϕ, is the angle between the reference plane formed by v_{sk}, v_{tk} and o, and the plane spanned by v_{sk}, v_{tk} and q.

ϕ is needed to uniquely locate every node in 3D space within SCS, since by only constraining d and θ there still is a degree of freedom, moving along a perimeter around s_k.

Hence, the message-passing is based on quadruplets of atoms - two atoms for interaction (s_k, r_k) and two atoms for direction (o, q).

To work with quadruplets, we make sure all angles are well defined and exclude overlapping atom quadruplets,

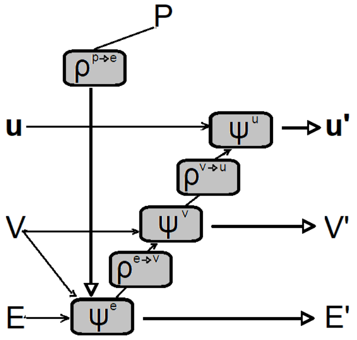

Fig. 3. Illustration of the information flow through GMP

like $a \neq b \neq c \neq d$ and polynomial radial envelopes are used to ensure smooth differentiability of the representations.

For GMP, computing distance d and θ between neighbouring nodes are straightforward. The torsional angles ϕ is calculated by projecting the neighbouring nodes $\mathcal{N}_{s_k}$ onto plane perpendicular to e_k and intersects with v_{s_k}. It reduces the computation space to the plane with v_{s_k} at the origin, and the torsional angles are the angles between q and o, where o is the last visited node meanwhile sweeping 2π radians, in a predefined direction.

Furthermore, by using the geometric representation $\psi(d, \theta, \phi)$ instead of P in GMP, it is possible to retain the positional information by only aggregating $\rho^{p\to e}$ of the three positional aggregators, illustrated in Fig. 3.

Therefore, we can realize Eq. (3) as the following:

$$\begin{aligned} e_k^{'} &= \Psi^e(e_k, v_{r_k}, v_{s_k}, \boldsymbol{E}_{s_k}, \rho^{p\to e}(\{t_h\}_{h=t_k \cup s_k \cup \mathcal{N}_{s_k}})), \\ v_i^{'} &= \Psi^v(v_i, \rho^{e\to v}(\boldsymbol{E}_i)), \\ u^{'} &= \Psi^u(\boldsymbol{u}, \rho^{v\to u})(\boldsymbol{V}^{'}) \end{aligned} \tag{7}$$

VI. DPGN: Deterministic point-cloud graph neural network

DPGN. The deterministic point-cloud graph neural network (DPGN) design is based upon a streamlined version of the MPNet architecture, and the architecture is illustrated in fig. (4).

DPGN was developed on the U0 target within the QM9 data set but generalizes to the other properties within QM9 and other data sets such as MD17 without architectural changes. The proposed changes to the architecture either improve the models' performance or reduces the models' complexity. We show the impact of the most relevant changes via the ablation studies in Sec.VII-E.

Embedding. The embedding block structure used in this paper is developed in Klicpera et al. [9]. It generates the initial message for the edge k, where Atomic numbers are represented by a learnable, randomly initialized embedding $h^0_{r_k}$, which is shared across molecules, via.

$$m^{(1)}_{s_k r_k} = \sigma([\boldsymbol{h}^{(0)}_{s_k} \| \boldsymbol{h}^{(0)}_{r_k} \| \boldsymbol{e}^{(k)}_{RBF_l}]\boldsymbol{W} + \boldsymbol{b}) \tag{8}$$

where $\|$ denotes concatenation, the local RBF embedding uses $c_{emb} = 5.0$ and the weight matrix $\boldsymbol{W}$ and bias $\boldsymbol{b}$ are learnable.

Interaction block. The embedding block is followed by multiple stacked interaction blocks. Each interaction block updates the message e_k, based upon the geometric message-passing as described in Sec. V-A geometric message-passing scheme. The interaction blocks input consists of the messages stored in the geometric embeddings $\tilde{e}_{RBF,n}(d), \tilde{a}_{CBF,ln}(d,\theta), \tilde{e}_{SBF,lmn}(d,\theta,\phi)$ in Eqs. (5)(6)(7), based on edge k and its neighbouring edges. The initial sizes for them are $N_{SHBF}, N_{SRBF} \times N_{SHBF}, N^2_{SRBF} \times N_{SHBF}$, respectivly. It also has the message e_k and the set of messages E_{s_k} that point to node V_{s_k} as inputs. The feature vector for the receiver node V_{r_k} for the edge k, is obtained by applying a permutation invariant function $\rho^{e\to v}$ to aggregate the messages from every neighbouring edge onto the receiver node. Finally, the updated global feature vector $\boldsymbol{u}^{'}$ is computed by applying a permutation invariant function $\rho^{v\to u}$ of the updated state of the nodes $\boldsymbol{V}^{(')}$

Bilinear Layer. DPGN use a bilinear layer, to give the network a more expressive transformation when modelling the complex interaction between the embeddings from SBF and CBF instead of a Hadamard product, which uses $N_{bilinear}FF$-dimensional weight tensor. Unfortunately, it is an expensive operation, but we think the accuracy improvement outweighs the price, see the table III to see its contribution.

Improving DPGNs expressivity. DPGNs expressivity is improved by using a power-sum multi-symmetric polynomial representation within the interaction block after the residual skip connections. This is done by adding a simple 3-WL discrimination power block (P-block) based upon Maron et als findings [45].

The expressivity is measured based on the Weisfeiler Lehman tests (k-WL), which is a hierarchy of increasingly powerful graph isomorphism tests, and a message-passing GNN cannot distinguish between graphs that are indistinguishable by the 1-WL test [26, 46].

The input tensor to the P-block is the message e^k after the residual blocks. Three multi layer perceptrons (MLPs) $m_1, m_2 : \mathbb{R}^a \to \mathbb{R}^b, m_3 : \mathbb{R}^a \to \mathbb{R}^{b'}$ to the input tensor, $m_l(e_k), l \in [\boldsymbol{3}]$. A matrix multiplication is

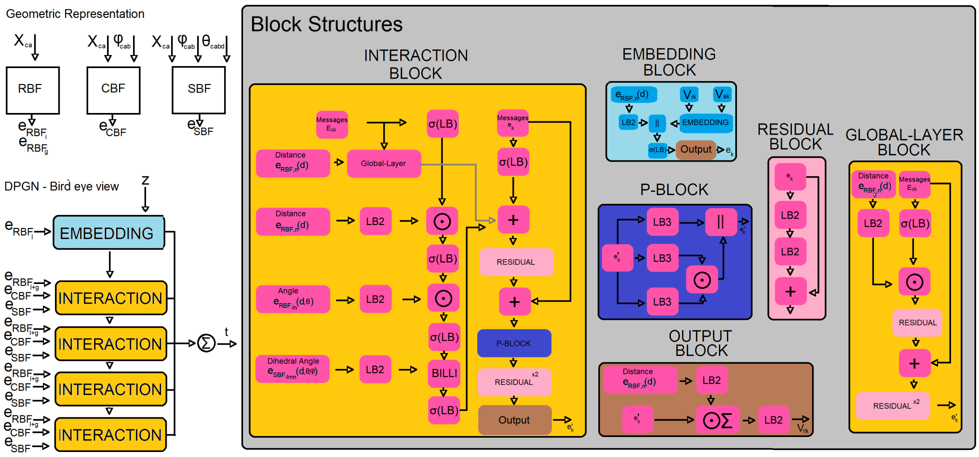

Fig. 4. Architecture of DPGN. On the left is a bird-eye view of the model. Where P gets transformed into a geometric representation, then gets fed into the interaction layers. The grey block shows the inner workings of the blocks. As for the symbolic meanings: $||$ denotes concatenation, $\odot$ denotes element-wise multiplication. $\sigma(LB)$ denotes a linear layer with a activation function. LB2 and LB3 denote a linear block of two and three linear layers. To reduce the effect of bottlenecks, every LB2 block gets down-projected, followed by up-projection.

performed between m_1 and m_2, then the output tensor gets concatenated with m_3

Continuous Differentiability. To retain the ability to use the architecture across various domains without modifications. The model has to be twice continuously differentiable to satisfy the conservation of energy. This is needed for molecular dynamics (MD) simulations since the force field is a conservative vector field. Therefore, DPGN like Klicpera et al. [9] uses the Swish activation function $\sigma(x) = x$, instead of a regular ReLU activation function. Additionally, the radial basis functions $\tilde{e}_{RBF}(d)$ is multiplied with a envelope function $u(d)$ that has a multiplicity of 3 at the cutoff c. Finally, this restriction limits the models to only use the atomic types and positional information and not use any auxiliary data.

Limitations. During the development of DPGN, it has solely focused on regression against QM9 property $U(0)$ the atomization energy at 0 kelvin (K). Experimentally, it has been shown to perform on par with state of the art models for various other properties within the same data set. It is also on par with state of the art models within molecular simulation based on the MD17 data set. We do not make any statement regarding its performance beyond this scope.

The addition of torsional forces to the message-passing scheme and the bilinear layer introduces significant computational overhead. This effect is partially mitigated by down-projecting the embeddings before entering these expensive operations. This model has not been able to beat MPNet on all the metrics, whereas MPNet is an ablated version of DPGN. This suggests that the expressivity of a model without torsional information and P-blocks is enough for some practical use cases. Whereas this more complex message-passing scheme evidently, gives an advantage when the task at hand is more challenging. This can be seen in the molecular dynamics (MD17) results, compared against DimeNet and SphereNet, which also are ablated versions of DPGN.

VII. Experiments

In our experiments, we evaluate the proposed modification on the MD17 data set for force fields and the 12 commonly used properties in the QM9 data set.

This experimental setting is chosen since there already exists a well-tuned baseline for reliable comparisons. 7 state-of-the-art baseline models are included for comparison. We assume that the results presented in the baseline models are reliable since we have not trained the models to validate their findings.

A. *QM9 Data set*

QM9 is a collection of electronic, energetic, geometric and thermodynamic properties of 134k stable small organic molecules made up of carbon, hydrogen, oxygen, nitrogen, fluorine (CHONF) calculated at the B3LYP/6-31G (2df,p) level of quantum chemistry. These molecules correspond to the subset of all 133,885 species with up to nine heavy atoms (CONF) out of the GDB-17 chemical universe of 166 billion organic molecules [47]. The property values were calculated using density functional theory (DFT) which takes in $\{r_a, Z_a\}$ to estimate the geometries minimal in energy, corresponding harmonic frequencies, dipole moments, polarizabilities, along with energies, enthalpies, and free energies of atomization.

Within the field of quantum chemistry, it is known as QM9 and has been a common benchmark data set, especially for data-driven methods which operate on graphs and point geometries [2, 3, 8]–[10, 46, 48, 49]. The data set was developed with the intent to serve the benchmarking of existing methods, development of new methods within the domain of quantum mechanics and machine learning, and systematic identification of structure-property relationships [47]. For a more detailed description of these properties, see appendix IX-A. The data set was obtained from https://bit.ly/3DT0IqU.

B. MD17 Data set

MD17 is a data set of configurations a molecular dynamics simulation of eight small thermalized organic molecules at $T = 500K$ and is computed at the PBE+vdW-TS level of electronic structure theory, whereas the goal of the benchmark is to predict both the energy and atomic forces considering one molecule at the time. The data set was obtained from http://quantum-machine.org/gdml/datasets.

C. Experimental Setup

The model architecture has been developed up against the QM9 validation set. We use 4 stacked interaction blocks with an embedding size of $F = 128$, embedding size of $F = 256$ for the output block, embedding size of $F = 64$ for the interaction block, embedding size of $F = 8$ while calculating the Bessel and spherical harmonic bases. For the basis functions we choose $N_{SHBF} = N_{CHBF} = 7$ and $N_{SRBF} = N_{CRBF} = N_{RBF} = 6$, for the weight tensor of the bilinear layer in the interaction block we use $N_{bilinear,SBF} = 32$. For the cutoff radiuses were $c = 5\mathring{A}$ for the interactions, $c = 7\mathring{A}$ for the global cutoff and $c = 10\mathring{A}$ for the other embeddings and the learning rate was $5x10^{-4}$.

D. Results.

QM9. We apply DPGN to the QM9 data set. The data set is split following the baselines, 110000 molecules for training, 10000 for validation and 13885 for testing. A new model was trained for all twelve target attributes separately, using mini-batches of 32 molecules at the time.

The experimental setups were optimized by minimizing the mean absolute error (MAE) loss using the Adam optimizer, and were tuned on their validation sets and then applied to the test sets. We compare our results with the results reported by several state-of-the-art models: SchNet [3], MGCN [50], DimeNet++ [51], SphereNet [17], MPNet [34], reported in Table I. As can be seen, DPGN is the best performing model on the property μ and is overall able to improve the results of the predecessor MPNet on six targets.

TABLE I
COMPARISON BETWEEN DPGN AND OTHER MODELS IN TERMS OF MAE ON QM9. ADDITIONAL DETAILS FOR THE PROPERTIES CAN BE FOUND IN APPENDIX IX-A. THE STATE OF THE ART RESULTS ARE HIGHLIGHTED IN BOLD WHILE SECOND-BEST RESULTS ARE UNDERLINED.

Target	SchNet	MGCN	DimeNet++	SphereNet	MPNet	DPGN
μ	0.033	0.0560	0.0297	0.0269	0.0329	**0.0267**
α	0.235	0.0300	0.0435	0.0465	**0.0270**	0.0530
ϵHOMO	41	42.1	24.6	**23.6**	26.5	26.6
ϵLUMO	34	57.4	19.5	18.9	**7.9**	21.6
$\Delta\epsilon$	63	64.2	32.6	32.3	**25.7**	52.7
<R2>	**0.073**	0.110	0.331	0.292	0.395	0.288
ZPVE	1.7	**1.12**	1.21	**1.12**	1.42	1.36
U_0	14	12.9	6.32	**6.26**	9.39	8.76
U	19	14.4	**6.28**	7.33	8.76	8.95
H	14	14.6	6.53	**6.40**	9.58	8.88
G	14	16.2	**7.56**	8.0	10.2	9.85
C_v	0.033	0.031	0.023	0.0215	**0.021**	0.027

MD17. The MD17 data set is used to examine DPGNs applicability within the domain of molecular dynamics simulations. The data set is split following the baseline, where it uses 1000 samples for training with a mini-batch size of 4, 10000 for validation and 122770 for testing with a mini-batch of 64.

The experimental setup is identical to the QM9 experiment. The results are compared with the results reported by several state-of-the-art models: SchNet [3], DimeNet [9], SphereNet [17], as can be seen in Table II:

TABLE II
COMPARISONS BETWEEN DPGN AND OTHER STATE-OF-THE-ART MODELS IN TERMS OF MAE OF FORCES ON MD17. THE BEST RESULTS ARE SHOWN IN BOLD AND THE SECOND-BEST RESULTS ARE SHOWN WITH UNDERLINES.

Molecule	sGDML	SchNet	DimeNet	SphereNet	DPGN
Aspirin	0.68	1.35	0.449	**0.430**	0.656
Benzene	0.20	0.31	0.187	0.178	**0.056**
Ethanol	0.33	0.39	0.230	0.208	**0.148**
Malonaldehyde	0.41	0.66	0.383	0.340	**0.245**
Naphthalene	**0.11**	0.58	0.215	0.178	0.178
Salicylic acid	**0.28**	0.85	0.374	0.360	0.500
Toluene	**0.14**	0.57	0.216	0.155	0.166
Uracil	0.24	0.56	0.301	0.267	**0.220**

Compared to the baselines, we can observe that DPGN can be used on molecular dynamics simulations without modifications to the architecture. It outperforms the other models by a large margin on Benzene, Ethanol, and Malonaldehyde. DPGN is the only model in this comparison, which has been able to beat sGDML on Uracil.

sGDML is the model within the original work that created the MD17 data set, it uses carefully designed auxiliary features to achieve its results. Whereas, the SchNet, DimeNet, SphereNet, and DPGN do not use any auxiliary information for their experiments. Hence, sGDML has poorer generalization power compared to the other baselines when it comes to molecular dynamics data set without hand-engineered features.

All the experiments are done on either a NVIDIA Tesla V100 GPU (32 GB) or an Titan RTX (24 GB).

E. Ablation Study.

The proposed UMP considers distance, angle and torsional angles and long-range forces within its representation. It leads to a complete representation of 3D information in addition to a sense of long-range force interactions to increase the models' expressivity.

In this section, we investigate the proposed architectural improvements on the target U_0. The effect of the proposed changes can be seen in Table III.

TABLE III
ABLATION STUDIES ON U0. FORCE MAE IN MEV AFTER 300 EPOCHS ON 30000 TRAINING SAMPLES, VALIDATED ON 1000 AND TESTED AGAINST 1000. ALL PROPOSED COMPONENTS YIELD IMPROVEMENTS.

Model	MAE
MPNet without modifications	0.122
Remove global effect when init embeddings	0.117
Adding torsional information (quadruplets)	0.113
Using bilinear layer instead of hadamard product when computing the effect of torsional forces	0.112
Adding P-block after residual layers	0.109

These results demonstrate the effectiveness of the proposed changes for realising the DPGN framework.

VIII. CONCLUSION

In this work, we have introduced DPGN which is a redefined architecture based on MPNet and 3DGN framework. To achieve this, a generic framework based on 3DGN is presented to provide a clear interface for 3D point cloud data sets. To realize the DPGN, the message-passing architecture UMP is presented which takes account of local and non-local energies by taking torsional forces into account.

By incorporating these modifications in the DPGN architecture, which can improve the errors on various targets within the QM9 data set.

Additionally, we have shown that the various modifications to the architecture, leads to significant performance within molecular dynamics simulations, without modifying the architecture.

Hence, the proposed modifications can contribute to the groundwork for developing a complete pipeline for analysing molecular systems.

ACKNOWLEDGEMENTS

The author would like to thank Yang Bin and Zhang Dalin for the supervision in addition to Klicpera Johannes for helpful discussions.

REFERENCES

[1] Z. Shui and G. Karypis, "Heterogeneous molecular graph neural networks for predicting molecule properties," in *2020 IEEE International Conference on Data Mining (ICDM)*. Los Alamitos, CA, USA: IEEE Computer Society, 2020, pp. 492–500.

[2] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," *34th International Conference on Machine Learning, ICML 2017*, vol. 3, pp. 2053–2070, 2017.

[3] H. E. Sauceda and A. Tkatchenko, "Schnet – a deep learning architecture for molecules and materials," no. June, p. 12, 2018.

[4] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, C. Gulcehre, F. Song, A. Ballard, J. Gilmer, G. Dahl, A. Vaswani, K. Allen, C. Nash, V. Langston, C. Dyer, N. Heess, D. Wierstra, P. Kohli, M. Botvinick, O. Vinyals, Y. Li, and R. Pascanu, "Relational inductive biases, deep learning, and graph networks," pp. 1–40, 2018. [Online]. Available: http://arxiv.org/abs/1806.01261

[5] L. LIU, T. Zhou, G. Long, J. Jiang, and C. Zhang, "Learning to propagate for graph meta-learning," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019.

[6] R. J. L. Townshend, R. Bedi, P. A. Suriana, and R. O. Dror, "End-to-end learning on 3d protein structure for interface prediction," 2019. [Online]. Available: https://github.com/drorlab/DIPS.

[7] S. Axelrod and R. G. Omez-Bombarelli, "Geom: Energy-annotated molecular conformations for property prediction and molecular generation," 2020.

[8] F. B. Fuchs, D. E. Worrall, V. Fischer, and M. Welling, "SE(3)-Transformers: 3D Roto-Translation Equivariant Attention Networks," no. 3, 2020. [Online]. Available: http://arxiv.org/abs/2006.10503

[9] J. Klicpera, J. Groß, and S. Günnemann, "Directional Message Passing for Molecular Graphs," *International Conference on Learning Representations (ICLR)*, no. 1, pp. 1–13, 2020. [Online]. Available: https://www.in.tum.de/daml/dimenet/

[10] Z. Qiao, M. Welborn, A. Anandkumar, F. R. Manby, and T. F. Miller, "OrbNet: Deep Learning for Quantum Chemistry Using Symmetry-Adapted Atomic-Orbital Features," 2020. [Online]. Available: http://arxiv.org/abs/2007.08026

[11] D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. J. Brostow, "Harmonic deep: Networks translation and rotation equivariance," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 7168–7177, 2017.

[12] M. Weiler, M. Geiger, M. Welling, W. Boomsma, and T. Cohen, "3d steerable cnns: Learning rotationally equivariant features in volumetric data," *Advances in Neural Information Processing Systems*, vol. 2018-Decem, no. Nips, pp. 10 381–10 392, 2018.

[13] R. Zhang, "Making convolutional networks shift-invariant again," 2019. [Online]. Available: https://richzhang.github.io/antialiased-cnns/.

[14] T. S. Cohen and M. W. WELLING, "Group equivariant convolutional networks," 2016.

[15] T. S. Cohen, M. Geiger, J. Köhler, and M. Welling, "Spherical cnns," 2018,.

[16] T. S. Cohen, M. Weiler, B. Kicanaoglu, and M. Welling, "Gauge equivariant convolutional networks and the icosahedral cnn," 2019.

[17] Y. Liu, L. Wang, M. Liu, X. Zhang, B. Oztekin, and S. Ji, "Spherical message passing for 3d graph networks," 2021.

[18] F. Noé, A. Tkatchenko, K. R. Müller, and C. Clementi, "Machine learning for molecular simulation," *Annual Review of Physical Chemistry*, vol. 71, pp. 361–390, 2020.

[19] I. Baskin, V. Palyulin, and N. Zefirov, "A neural device for searching direct correlations between structures and properties of chemical compounds," *Journal of Chemical Information and Modeling*, vol. 37, pp. 715–721, 1997.

[20] A. Sperduti and A. Starita, "Supervised neural networks for the classification of structures," *IEEE Transactions on Neural Networks*, vol. 8, no. 3, pp. 714–735, 1997.

[21] F. Scarselli, S. Yong, M. Gori, M. Hagenbuchner, A. Tsoi, and M. Maggini, "Graph neural networks for ranking web pages." 2005, pp. 666–672.

[22] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2009. [Online]. Available: https://persagen.com/files/misc/scarselli2009graph.pdf

[23] D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams, "Convolutional networks on graphs for learning molecular fingerprints," in *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, Eds., vol. 28. Curran Associates, Inc., 2015.

[24] S. Kearnes, K. McCloskey, M. Berndl, V. Pande, and P. Riley, "Molecular graph convolutions: moving beyond fingerprints," *Journal of Computer-Aided Molecular Design*, vol. 30, no. 8, pp. 595–608, 2016.

[25] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2017.

[26] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" 2019.

[27] P. Veličkovi ́veličkovi ́c, G. Cucurull, A. Casanova, A. Romero, P. Lì, and Y. Bengio, "Graph attention networks," 2018.

[28] Y. Li, R. Zemel, M. Brockschmidt, and D. Tarlow, "Gated graph sequence neural networks," 2017.

[29] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," 2018. [Online]. Available: https://github.com/

[30] J. Behler and M. Parrinello, "Generalized neural-network representation of high-dimensional potential-energy surfaces," *Phys. Rev. Lett.*, vol. 98, p. 146401, Apr 2007. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.98.146401

[31] A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi, "Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons," *Phys. Rev. Lett.*, vol. 104, p. 136403, Apr 2010. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.104.136403

[32] O. T. Unke and M. Meuwly, "Physnet: A neural network for predicting energies, forces, dipole moments and partial charges," 2019.

[33] C. W. Park, M. Kornbluth, J. Vandermause, C. Wolverton, B. Kozinsky, and J. P. Mailoa, "Accurate and scalable multi-element graph neural network force field and molecular dynamics with direct force architecture," 2020.

[34] S. Jakup, "Unveiling the predictive power of the 3d roto-translation equivariant graph neural network mpnet," 2021.

[35] C. Esteves, C. Allen-Blanchette, A. Makadia, and K. Daniilidis, "Learning so(3) equivariant representations with spherical cnns," 2018. [Online]. Available: http://github.com/daniilidis-group/spherical-cnn

[36] N. C. Thomas, T. Smidt, S. Kearnes, L. Yang, L. Li, K. Kohlhoff, and P. Riley, "Tensor field networks: Rotation- and translation-equivariant neural networks for 3d point clouds," 2018.

[37] B. Anderson, T.-S. Hy, and R. Kondor, "Cormorant: Covariant molecular neural networks," 2019.

[38] M. Finzi, S. Stanton, P. Izmailov, and A. G. Wilson, "Generalizing convolutional neural networks for equivariance to lie groups on arbitrary continuous data," 2020. [Online]. Available: https://github.com/mfinzi/LieConv

[39] S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt, and B. Kozinsky, "Se(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials," 2021.

[40] N. Dym and H. Maron, "On the Universality of Rotation Equivariant Point Cloud Networks," pp. 1–20, 2020. [Online]. Available: http://arxiv.org/abs/2010.02449

[41] J. Klicpera, F. Becker, and S. Günnemann, "Gemnet: Universal directional graph neural networks for molecules," 2021.

[42] T. Xie and J. C. Grossman, "Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties," *Phys. Rev. Lett.*, vol. 120, p. 145301, 2018. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.120.145301

[43] Z. Wu, B. Ramsundar, E. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, and V. Pande, "Moleculenet: a benchmark for molecular machine learning," *Chem. Sci.*, vol. 9, pp. 513–530, 2018. [Online]. Available: http://dx.doi.org/10.1039/C7SC02664A

[44] S. Ravanbakhsh, J. Schneider, and B. Póczos, "Equivariance through parameter-sharing," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. International Convention Centre, Sydney, Australia: PMLR, 2017, pp. 2892–2901. [Online]. Available: http://proceedings.mlr.press/v70/ravanbakhsh17a.html

[45] H. Maron, H. Ben-Hamu, H. Serviansky, and Y. Lipman, "Provably powerful graph networks," 2020.

[46] C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan, and M. Grohe, "Weisfeiler and leman go neural: Higher-order graph neural networks," *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, pp. 4602–4609, 2019.

[47] R. Ramakrishnan, P. O. Dral, M. Rupp, and O. A. Von Lilienfeld, "Quantum chemistry structures and properties of 134 kilo molecules," *Scientific Data*, vol. 1, no. 1, pp. 1–7, 2014. [Online]. Available: www.nature.com/sdata/

[48] M. Finzi, S. Stanton, P. Izmailov, and A. G. Wilson, "Generalizing Convolutional Neural Networks for Equivariance to Lie Groups on Arbitrary Continuous Data," 2020. [Online]. Available: http://arxiv.org/abs/2002.12880

[49] N. W. Gebauer, M. Gastegger, and K. T. Schütt, "Symmetry-adapted generation of 3D point sets for the targeted discovery of molecules," *Advances in Neural Information Processing Systems*, vol. 32, no. NeurIPS, 2019.

[50] Y. Wang, J. Ren, K. D.-M. Yan, D.-M. Yan, . X. Zhang, J. Guo, and X. Zhang, "Mgcn: Descriptor learning using multiscale gcns," *ACM Trans. Graph. 39, 4, Article*, vol. 122, p. 15, 2020. [Online]. Available: https://doi.org/10.1145/3386569.3392443

[51] J. Klicpera, S. Giri, J. T. Margraf, and S. Günnemann, "Fast and Uncertainty-Aware Directional Message Passing for

Non-Equilibrium Molecules," Tech. Rep. [Online]. Available: https://ml4molecules.github.io

IX. Appendices

In this supplementary material we provide more details about the quantum properties in the QM9 data set. The quantum property descriptions are taken from [34]

A. Description of the Quantum Properties

$\boldsymbol{U_0}$ (eV): is the atomization energy at 0 kelvin (K). This is the energy required to break up a molecule to its constituent atoms at a temperature of $0K$. The calculation assumes that the molecule is held at a fixed volume.

$\boldsymbol{U}$ (eV): is the atomization energy at room temperature.

$\boldsymbol{H}$ (eV): is the enthalpy of atomization at room temperature. It is in similar spirit to the spirit of U, without depending on the assumption that the volume is fixed.

$\boldsymbol{G}$ (eV): is the free energy of atomization. It also is in similar spirit to G and U, but with the assumption that the system is held in a fixed temperature and pressure.

$\boldsymbol{\omega_1}$ (cm^{-1}): is the molecules highest fundamental frequency. Every molecule has fundamental vibrational modes. ω is the mode that requires the most energy.

$\boldsymbol{ZPEV}$ (eV): is the zero point vibrational energy, which is the vibrational mode of the molecule at $0K$.

$\boldsymbol{\epsilon_{HOMO}}$ (eV): is the energy of the highest occcupied electronic state.

$\boldsymbol{\epsilon_{LUMO}}$ (eV)): is the energy of the lowest electronic state that is unoccupied.

$\boldsymbol{\Delta_\epsilon}$ (eV)): is the difference between LUMO and HOMO. It is the energy transition that can occur when an electron is excited from an occupied state to an unoccupied state.

$\boldsymbol{\langle R^2 \rangle}$ $(Bohr^2)$: is the second moment of the charge distribution. Also known as the electronic spatial extent.

$\boldsymbol{\mu}$ (Debye): is the norm of the dipole moment. It is related to how anisotropically the charge is distributed. It says something about the strength of the field far from the molecule.

$\boldsymbol{\alpha}$ $(Bohr^3)$: is the static polarizability. It measures the extent to which a molecule can spontaneously incure a dipole moment in response to an external field [2].