

Equivariant Graph Neural Networks for 3D Macromolecular Structure

Bowen Jing¹ Stephan Eismann^{1,2} Pratham N. Soni¹ Ron O. Dror¹

Abstract

Representing and reasoning about 3D structures of macromolecules is emerging as a distinct challenge in machine learning. Here, we extend recent work on *geometric vector perceptrons* and apply equivariant graph neural networks to a wide range of tasks from structural biology. Our method outperforms all reference architectures on three out of eight tasks in the ATOM3D benchmark, is tied for first on two others, and is competitive with equivariant networks using higher-order representations and spherical harmonic convolutions. In addition, we demonstrate that transfer learning can further improve performance on certain downstream tasks. Code is available at <https://github.com/drorlab/gvp-pytorch>.

1. Introduction

Learning on 3D structures of macromolecules (such as proteins) is a rapidly growing area of machine learning with promising applications but also domain-specific challenges. In particular, methods should possess an efficient and precise representation of structures with thousands of atoms and faithfully reason about their 3D geometry independent of orientation and position (Laine et al., 2021).

Equivariant neural networks (ENNs), operating on point-cloud representations of structures with geometric vector/tensor features and spherical harmonic convolutions, address these challenges. These networks have shown promising results on small molecule datasets (Thomas et al., 2018; Kondor, 2018; Anderson et al., 2019; Fuchs et al., 2020; Batzner et al., 2021) and recently also in the context of learning from macromolecular structure (Eismann et al., 2020a;b; Townshend et al., 2020).

Graph neural networks (GNNs) have enjoyed widespread

¹Department of Computer Science, Stanford University, USA ²Department of Applied Physics, Stanford University, USA. Correspondence to: Bowen Jing <bjing@cs.stanford.edu>, Stephan Eismann <seismann@cs.stanford.edu>, Ron O. Dror <rondror@cs.stanford.edu>.

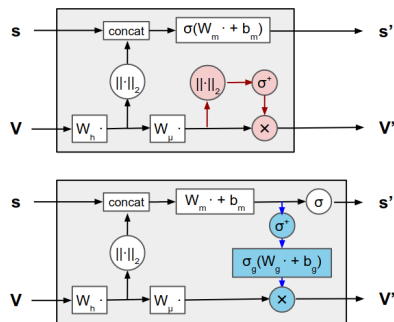


Figure 1. Schematic of the original geometric vector perceptron (GVP) as described in Jing et al. (2021) (top) and the modified GVP presented in Algorithm 1 (bottom). The original vector nonlinearity (in red) has been replaced with vector gating (in blue), allowing information to propagate from the scalar channels to the vector channels. Circles denote row- or element-wise operations. The modified GVP is the core module in our equivariant GNN.

use on macromolecular structure due to the natural representation of structures as graphs, with residues or atoms as nodes and edges drawn based on bonding or spatial proximity (Ingraham et al., 2019; Baldassarre et al., 2020). However, these networks generally indirectly encode the 3D geometry in terms of pairwise distances, angles, and other scalar features. Equivariant message-passing, proposed by Jing et al. (2021) and more recently Satorras et al. (2021) and Schütt et al. (2021), seeks to instead incorporate the equivariant representations of ENNs within the message-passing framework of GNNs. This presents an alternative, mathematically simpler approach to equivariance in lieu of the equivariant convolutions of ENNs, and also has the advantage of leveraging the relational reasoning of GNNs.

Here, we present a GNN with equivariant message-passing and demonstrate that it achieves strong results across a wide variety of tasks on macromolecular structures. Our architecture extends GVP-GNNs (Jing et al., 2021) and we evaluate its performance on ATOM3D (Townshend et al., 2020), a comprehensive suite of tasks and datasets from structural biology. We also demonstrate that our architecture can leverage transfer learning—i.e., pre-training on a data-rich task to improve performance on a data-poor task. This is particularly desirable for machine learning on macromolecules, as structural data is often expensive and difficult to obtain.

2. Equivariant Graph Neural Networks

GVP-GNNs (Jing et al., 2021) are equivariant graph neural networks in which all node and edge embeddings are tuples (\mathbf{s}, \mathbf{V}) of scalar features $\mathbf{s} \in \mathbb{R}^n$ and geometric vector features $\mathbf{V} \in \mathbb{R}^{\nu \times 3}$. Message and update functions are parameterized by *geometric vector perceptrons* (GVPs)—modules mapping between tuples (\mathbf{s}, \mathbf{V}) while preserving rotation equivariance. Here, we extend GVP-GNNs—originally designed for representation of protein structures at the amino acid residue level—to atomic-level structure representations. This enables the architecture to be applied to a larger variety of tasks, and motivates *vector gating* as an adaptation for the change in input representation.

Vector gating In the originally defined GVP, the vector outputs are functions of the vector inputs, but not the scalar inputs—i.e., vector features at all stages of message-passing are independent of residue identity or other scalar features. This is not a significant issue for residue-level structure graphs, in which nodes start with vector features in the form of residue orientations. However, in atomic-level structure graphs, individual atoms do not necessarily start with an orientation. We therefore introduce *vector gating* to propagate information from the scalar channels into the vector channels (Algorithm 1). In this operation, the scalar features s_m are transformed and passed through a sigmoid activation σ_g in order to “gate” the vector output \mathbf{V}' , replacing the vector nonlinearity. Because s_m is invariant and the gating is row-wise, the equivariance of \mathbf{V}' is unaffected.

Algorithm 1 Geometric vector perceptron (with vector gate)

Input: Scalar and vector features $(\mathbf{s}, \mathbf{V}) \in \mathbb{R}^n \times \mathbb{R}^{\nu \times 3}$.
Output: Scalar and vector features $(\mathbf{s}', \mathbf{V}') \in \mathbb{R}^m \times \mathbb{R}^{\mu \times 3}$.
 $h \leftarrow \max(\nu, \mu)$ (or separately specified)
GVP:
 $\mathbf{V}_h \leftarrow \mathbf{W}_h \mathbf{V} \in \mathbb{R}^{h \times 3}$
 $\mathbf{V}_\mu \leftarrow \mathbf{W}_\mu \mathbf{V}_h \in \mathbb{R}^{\mu \times 3}$
 $s_h \leftarrow \|\mathbf{V}_h\|_2$ (row-wise) $\in \mathbb{R}^h$
 $s_{h+n} \leftarrow \text{concat}(s_h, \mathbf{s}) \in \mathbb{R}^{h+n}$
 $s_m \leftarrow \mathbf{W}_m s_{h+n} + \mathbf{b}_m \in \mathbb{R}^m$
 $\mathbf{s}' \leftarrow \sigma(s_m) \in \mathbb{R}^m$
 $\mathbf{V}' \leftarrow \sigma_g(\mathbf{W}_g[\sigma^+(s_m)] + \mathbf{b}_g) \odot \mathbf{V}_\mu$ (row-wise) $\in \mathbb{R}^{\mu \times 3}$
return $(\mathbf{s}', \mathbf{V}')$

The modification enables the GVP to inherit the Universal Approximation Property of dense layers with respect to rotation- and reflection-equivariant functions $F: \mathbb{R}^{\nu \times 3} \rightarrow \mathbb{R}^3$, in addition to the approximation property for *invariant* functions shown by Jing et al. (2021).

Theorem. *Let R describe an arbitrary rotation and/or reflection in \mathbb{R}^3 and G_s be the vector outputs of a GVP defined with $n = 0$, $\mu = 6$, and sigmoidal σ, σ^+ . For $\nu \geq 3$ let $\Omega^\nu \subset \mathbb{R}^{\nu \times 3}$ be the set of all $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_\nu]^T \in \mathbb{R}^{\nu \times 3}$ such that $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ are linearly independent and $0 <$*

$\|\mathbf{v}_i\|_2 \leq b$ for all i and some finite $b > 0$. Then for any continuous $F: \Omega^\nu \rightarrow \mathbb{R}^3$ such that $(F \circ R)(\mathbf{V}) = (R \circ F)(\mathbf{V})$ and for any $\epsilon > 0$, there exists a form $f(\mathbf{V}) = \mathbf{1}^T G_s(\mathbf{V})$ such that $|F(\mathbf{V})_i - f(\mathbf{V})_i| < 6bC\epsilon$ (with finite $C > 0$) for all $i = 1, 2, 3$ and all $\mathbf{V} \in \Omega^\nu$.

See the Appendix for a proof. As a corollary, a GVP with nonzero n —that is, with scalar inputs—can approximate similarly defined functions over the full input domain $\mathbb{R}^n \times \mathbb{R}^{\nu \times 3}$, corresponding to the propagation of information from scalar to vector channels. Using these modified GVPs, we build GVP-GNN with the same formulation of message-passing and node update as in Jing et al. (2021).

3. ML on 3D Macromolecular Structure

ATOM3D (Townshend et al., 2020) is a collection of eight tasks and datasets for learning on atomic-level 3D molecular structure. These tasks span several classes of molecules (proteins, RNAs, small molecules, and complexes) and problem formulations (regression and classification) encountered in structural biology (Table 1). Townshend et al. (2020) established reference benchmarks for convolutional, graph, and equivariant neural networks—the three architectures most broadly applicable to molecular structure—on these tasks. We compare our modified equivariant GVP-GNN to these reference architectures.

The tasks in ATOM3D span several orders of magnitude in both dataset size and structure size (a rough proxy for task complexity), and therefore range from data-rich (SMP, RES) to data-poor (MSP, LEP). However, these tasks share a common underlying representation and problem domain, presenting an opportunity for *transfer learning* to improve performance on data-poor tasks. We therefore investigate if leveraging the learned parameters from the first two layers of the trained model on SMP (target μ) and RES can improve over a non-pretrained model on the other tasks.

Architecture We represent a macromolecular structure as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where each node $\mathbf{v}_i \in \mathcal{V}$ corresponds to an atom and is featured by a one-hot encoding of its element type. We draw edges $\mathcal{E} = \{\mathbf{e}_{j \rightarrow i}\}_{i \neq j}$ for all i, j whose pairwise distance is less than 4.5 Å. Each edge is featured with a unit vector in the direction of the edge and a Gaussian RBF encoding of its length. We use hidden embeddings with 16 vector and 100 scalar channels. For all tasks, we use a GVP-GNN with five layers, followed by a mean pool (unless otherwise noted) and two dense layers. For all GVPs, we use $\sigma = \text{ReLU}$ and $\sigma^+ = \text{id}$.

We do not encode any bond or bond type information, nor do we distinguish atoms from different molecules in the same structure. This is in order to standardize the architecture across tasks as much as possible, but may handicap our

Table 1. Tasks in ATOM3D (Townshend et al., 2020) encompass a broad range of structure types, problem formulations, dataset sizes, and structure sizes. PPI, RES, MSP, and LEP are classification tasks; the others are regression tasks. Further, PPI, MSP, and LEP have paired inputs, so methods for these tasks use weight-tying between two independent forward passes. Dataset size is the number of training examples, and structure size is the geometric mean of the number of nodes in training examples. In paired tasks, each pair is counted as two examples. Where necessary, we use dataloader conventions from (Townshend et al., 2020).

Name	Input Structure(s)	Prediction	Dataset Size	Structure Size
Small Molecule Properties (SMP)	Small molecule	Physiochemical property	104×10^3	18
Protein-Protein Interface (PPI)	Two protein chains + residue from each	Whether the residues come into contact when the parent chains interact	2.1×10^6	999
Residue Identity (RES)	Environment around a masked residue	Masked residue identity	1.0×10^6	574
Mutation Stability Prediction (MSP)	Protein complex + same complex with mutation	Whether the mutation stabilizes the complex	5728	3236
Ligand Binding Affinity (LBA)	Protein-ligand complex	Negative log affinity (pK_d)	3507	382
Ligand Efficacy Prediction (LEP)	Protein-ligand complex in active+inactive conformations	Whether the ligand activates the protein	608	3001
Protein Structure Ranking (PSR)	Protein	GDT-TS relative to native structure	25.4×10^3	1384
RNA Structure Ranking (RSR)	RNA	RMSD relative to native structure	12.5×10^3	2161

Table 2. Comparison of the GVP-GNN with the CNN, GNN, and ENN reference architectures on ATOM3D. The GVP-GNN is the best architecture on three out of eight tasks, more than any other method. For each metric, the best model is in bold. SMP metrics are MAE. Metrics are labeled with \uparrow/\downarrow if higher/lower is better, respectively. Results are mean \pm S.D. over three training runs.

Task	Metric	ATOM3D Reference			
		CNN	GNN	ENN	GVP-GNN
SMP	μ [D] \downarrow	0.754 ± 0.009	0.501 ± 0.002	0.052 ± 0.007	0.049 ± 0.002
	ϵ_{gap} [eV] \downarrow	0.580 ± 0.004	0.137 ± 0.002	0.095 ± 0.021	0.065 ± 0.001
	U_0^{at} [eV] \downarrow	3.862 ± 0.594	1.424 ± 0.211	0.025 ± 0.001	0.143 ± 0.007
PPI	AUROC \uparrow	0.844 ± 0.002	0.669 ± 0.001	—	0.866 ± 0.004
RES	accuracy \uparrow	0.451 ± 0.002	0.082 ± 0.002	0.072 ± 0.005	0.527 ± 0.003
MSP	AUROC \uparrow	0.574 ± 0.005	0.621 ± 0.009	0.574 ± 0.040	0.680 ± 0.015
LBA	RMSE \downarrow	1.416 ± 0.021	1.570 ± 0.025	1.568 ± 0.012	1.594 ± 0.073
LEP	AUROC \uparrow	0.589 ± 0.020	0.740 ± 0.010	0.663 ± 0.100	0.628 ± 0.055
	mean R_S \uparrow	0.431 ± 0.013	0.515 ± 0.010	—	0.511 ± 0.010
PSR	global R_S \uparrow	0.789 ± 0.017	0.755 ± 0.004	—	0.845 ± 0.008
	mean R_S \uparrow	0.264 ± 0.046	0.234 ± 0.006	—	0.211 ± 0.142
RSR	global R_S \uparrow	0.372 ± 0.027	0.512 ± 0.049	—	0.330 ± 0.054

method relative to those that do use this information.

In all tasks except SMP and LBA, we omit the hydrogen atoms. In the pairwise tasks PPI, LEP, and MSP, we perform independent forward passes and concatenate the outputs. In residue-specific tasks, we pick out the output embedding of the alpha carbon (PPI, RES), or mean pool over the residue (MSP). These conventions are from Townshend et al. (2020).

4. Results

We compare the vector-gated GVP-GNN with the reference six-layer CNN, five-layer GNN, and four-layer ENN architectures on the ATOM3D test sets (Table 2). The GVP-GNN achieves generally strong performance: it is the top method on three tasks (PPI, RES, MSP)—more than any other method—and tied for first on two (SMP, PSR). However, in a head-to-head comparison with the standard GNN,

Table 3. Transfer learning improves performance of GVP-GNNs on a number of ATOM3D tasks (PPI, MSP, PSR). For each task, we compare models pretrained on SMP and RES with the original models. Pretrained models that outperform the original models are shown in bold. The original model is shown in bold if neither pretrained model outperforms it. p -values are shown for a one-tailed t -test between the pretrained model and original model, and bolded if $p \leq 0.05$. Metrics are labeled with \uparrow/\downarrow if higher/lower is better. Results are mean \pm S.D. over three training runs.

Task	Metric	No Pretraining	SMP Pretraining	p	RES Pretraining	p
PPI	AUROC \uparrow	0.866 \pm 0.004	0.874 \pm 0.001	0.04	—	—
RES	accuracy \uparrow	0.527 \pm 0.003	0.531 \pm 0.001	0.07	—	—
MSP	AUROC \uparrow	0.680 \pm 0.015	0.711 \pm 0.007	0.03	0.709 \pm 0.014	0.04
LBA	RMSE \downarrow	1.594 \pm 0.073	1.649 \pm 0.118	0.73	1.676 \pm 0.162	0.76
LEP	AUROC \uparrow	0.628 \pm 0.055	0.567 \pm 0.210	0.67	0.466 \pm 0.110	0.95
PSR	mean R_S \uparrow	0.511 \pm 0.010	0.515 \pm 0.007	0.31	0.515 \pm 0.008	0.35
	global R_S \uparrow	0.845 \pm 0.008	0.848 \pm 0.006	0.29	0.862 \pm 0.006	0.02
RSR	mean R_S \uparrow	0.211 \pm 0.142	0.194 \pm 0.161	0.55	0.156 \pm 0.155	0.66
	global R_S \uparrow	0.330 \pm 0.054	0.059 \pm 0.303	0.87	0.308 \pm 0.072	0.65

the GVP-GNN scores higher on only 8 out of 12 metrics, suggesting that equivariant, vector-valued representations may not be equally useful for all datasets or metrics.

The reference ENN is a Cormorant neural network—a point cloud network with spherical harmonic convolutions (Anderson et al., 2019). GVP-GNN is on par with this ENN, scoring better on 4 out of 7 metrics. (The ENN was not trained on PPI, PSR, and RSR.) However, the ENN uses equivariant representations up to $L = 3$ —comparable to 3rd order geometric tensors—while the GVP-GNN only uses geometric *vectors*, or 1st order tensors. Thus, equivariant message-passing with lower order tensors is competitive with higher-order ENNs on molecular tasks.

Compared with the GNN and ENN collectively, the GVP-GNN falls short on the tasks involving ligand-protein complexes (LBA, LEP). We postulate that this may be because our general atomic-level input features omit information that explicitly distinguishes ligand atoms from protein atoms.

Transfer learning GVP-GNNs pretrained on data-rich tasks (SMP, RES) are able to improve performance on a number of downstream tasks (Table 3). Specifically, the SMP-pretrained model improves performance on PPI, RES, MSP, and PSR, and the RES-pretrained model improves performance on MSP and PSR. (The RES model was not fine-tuned on PPI because the datasets are of similar size.) Many of these improvements are statistically significant at the $\alpha = 0.05$ level. To the best of our knowledge, this is the first time transfer learning has been successfully demonstrated for machine learning methods operating on 3D representations of macromolecular structures.

We observe that even in cases where pretraining does not improve the performance, it still may expedite training. Figure 2 compares the learning curves for the original and

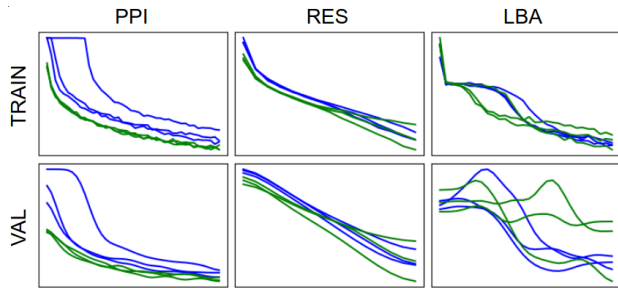


Figure 2. Learning curves (loss vs. epoch) for the original (blue) and SMP-pretrained (green) models on PPI, RES, LBA, all illustrating expedited training but with different final results. The losses have been Gaussian smoothed with $\sigma = 2$ epochs.

SMP-pretrained models on PPI, RES, and LBA. On all three tasks, the learned weights from SMP appear to help on the new dataset, as learning is expedited by several epochs. However, while for PPI and RES this head start leads to improved or comparable performance, for LBA it increases overfitting and hurts performance.

5. Conclusion

We have demonstrated the systematic application of equivariant graph neural networks to macromolecular structures. Our architecture extends GVP-GNN to atomic-level structures, endows it with additional universal approximation properties, and achieves strong performance across a wide range of structural biology tasks. We also show that leveraging pretrained equivariant representations can boost performance on downstream tasks. These results suggest that equivariant message-passing is a powerful and versatile paradigm for machine learning on molecules.

References

- Anderson, B., Hy, T.-S., and Kondor, R. Cormorant: Covariant molecular neural networks. *arXiv preprint arXiv:1906.04015*, 2019.
- Baldassarre, F., Menéndez Hurtado, D., Elofsson, A., and Azizpour, H. GraphQA: protein model quality assessment using graph convolutional networks. *Bioinformatics*, 2020.
- Batzner, S., Smidt, T. E., Sun, L., Mailoa, J. P., Kornbluth, M., Molinari, N., and Kozinsky, B. Se (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *arXiv preprint arXiv:2101.03164*, 2021.
- Eismann, S., Townshend, R. J., Thomas, N., Jagota, M., Jing, B., and Dror, R. O. Hierarchical, rotation-equivariant neural networks to select structural models of protein complexes. *Proteins: Structure, Function, and Bioinformatics*, 2020a. doi: 10.1002/prot.26033.
- Eismann, S., Suriana, P., Jing, B., Townshend, R. J., and Dror, R. O. Protein model quality assessment using rotation-equivariant, hierarchical neural networks. *arXiv preprint arXiv:2011.13557*, 2020b.
- Fuchs, F., Worrall, D., Fischer, V., and Welling, M. SE(3)-transformers: 3D roto-translation equivariant attention networks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1970–1981, 2020.
- Ingraham, J., Garg, V., Barzilay, R., and Jaakkola, T. Generative models for graph-based protein design. In *Advances in Neural Information Processing Systems*, pp. 15794–15805, 2019.
- Jing, B., Eismann, S., Suriana, P., Townshend, R. J. L., and Dror, R. Learning from protein structure with geometric vector perceptrons. In *International Conference on Learning Representations*, 2021.
- Kondor, R. N-body networks: A covariant hierarchical neural network architecture for learning atomic potentials. *arXiv preprint arXiv:1803.01588*, 2018.
- Laine, E., Eismann, S., Elofsson, A., and Grudin, S. Protein sequence-to-structure learning: Is this the end(-to-end revolution)? *arXiv preprint arXiv:2105.07407*, 2021.
- Satorras, V. G., Hoogeboom, E., and Welling, M. E(n) equivariant graph neural networks. *arXiv preprint arXiv:2102.09844*, 2021.
- Schütt, K. T., Unke, O. T., and Gastegger, M. Equivariant message passing for the prediction of tensorial properties and molecular spectra. *arXiv preprint arXiv:2102.03150*, 2021.
- Thomas, N., Smidt, T., Kearnes, S., Yang, L., Li, L., Kohlhoff, K., and Riley, P. Tensor field networks: Rotation-and translation-equivariant neural networks for 3D point clouds. *arXiv preprint arXiv:1802.08219*, 2018.
- Townshend, R. J., Vögele, M., Suriana, P., Derry, A., Powers, A., Laloudakis, Y., Balachandrar, S., Anderson, B., Eismann, S., Kondor, R., et al. ATOM3D: Tasks on molecules in three dimensions. *arXiv preprint arXiv:2012.04035*, 2020.

Appendix

Theorem. *Let R describe an arbitrary rotation and/or reflection in \mathbb{R}^3 and G_s be the vector outputs of a GVP defined with $n = 0$, $\mu = 6$, and sigmoidal σ, σ^+ . For $\nu \geq 3$ let $\Omega^\nu \subset \mathbb{R}^{\nu \times 3}$ be the set of all $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_\nu]^T \in \mathbb{R}^{\nu \times 3}$ such that $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ are linearly independent and $0 < \|\mathbf{v}_i\|_2 \leq b$ for all i and some finite $b > 0$. Then for any continuous $F : \Omega^\nu \rightarrow \mathbb{R}^3$ such that $(F \circ R)(\mathbf{V}) = (R \circ F)(\mathbf{V})$ and for any $\epsilon > 0$, there exists a form $f(\mathbf{V}) = \mathbf{1}^T G_s(\mathbf{V})$ such that $|F(\mathbf{V})_i - f(\mathbf{V})_i| < 6bC\epsilon$ (with finite $C > 0$) for all $i = 1, 2, 3$ and all $\mathbf{V} \in \Omega^\nu$.*

Proof. We start by writing out the GVP G_s in full as $G_s(\mathbf{V}) = \sigma_g(\mathbf{W}_g s^+ + \mathbf{b}_g) \odot \mathbf{W}_\mu \mathbf{W}_h \mathbf{V}$, where $s^+ = \sigma^+(\mathbf{W}_m \|\mathbf{W}_h \mathbf{V}\|_2 + \mathbf{b}_m)$ and $\|\cdot\|_2$ is a row-wise norm. The idea is to show that $\mathbf{W}_\mu \mathbf{W}_h$ can extract three linearly independent vectors (and their negatives), and that the vector gate can construct the coefficients on these vectors so that they sum to $F(\mathbf{V})$.

First, because $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ are linearly independent, we can write $F(\mathbf{V}) = \sum_{i=1}^3 c_i \mathbf{v}_i$ for some coefficients c_i dependent on \mathbf{V} . That is, c_i is a function $c_i(\mathbf{V})$. Let $C = \max_{i=1,2,3} \max_{\mathbf{V} \in \Omega^\nu} |c_i(\mathbf{V})|$. Now define $c_i^+ = \max(c_i, 0)/C$ and $c_i^- = -\min(c_i, 0)/C$ such that $c_i = C(c_i^+ - c_i^-)$. These functions c_i^+ and c_i^- are bounded between 0 and 1, so we can then define $\tilde{c}_i^+ = \sigma_g^{-1}(c_i^+)$ and correspondingly for \tilde{c}_i^- . These functions must be invariant under R , for otherwise F would not be equivariant under R .

Next, let \mathbf{W}_h be parameterized as described in [Jing et al. \(2021\)](#). Using the earlier result, there exists a form $\mathbf{w}_{i\pm}^T \sigma^+(\mathbf{W}_{m,i\pm} \|\mathbf{W}_h \mathbf{V}\|_2 + \mathbf{b}_m) \in \mathbb{R}$ parameterized by $\mathbf{w}_{i\pm}, \mathbf{W}_{m,i\pm}$ that ϵ -approximates each \tilde{c}_i^\pm . Then letting $\mathbf{W}_m = [\mathbf{W}_{m,1+}^T \quad \mathbf{W}_{m,1-}^T \quad \dots]$ and letting \mathbf{W}_g have the $\mathbf{w}_{i\pm}$ on the diagonals, we see that the entries of $\mathbf{W}_g s^+$ will ϵ -approximate the \tilde{c}_i^\pm . Now let $\mathbf{b}_g = \mathbf{0}$ such that the vector gate is $\sigma_g(\mathbf{W}_g s^+) \in \mathbb{R}^6$ where s^+ is as defined above. Because σ_g has less than unit derivative everywhere, we can see that the entries of $\sigma_g(\mathbf{W}_g s^+)$ will ϵ -approximate the bounded functions c_i^\pm .

Finally, let \mathbf{W}_μ be parameterized such that $\mathbf{W}_\mu \mathbf{W}_h = C \begin{bmatrix} I_3 & \dots \\ -I_3 & \dots \end{bmatrix}$. (It is straightforward to do this given the \mathbf{W}_h described above.) Then $\mathbf{W}_\mu \mathbf{W}_h \mathbf{V} = C [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, -\mathbf{v}_1, -\mathbf{v}_2, -\mathbf{v}_3]^T \in \mathbb{R}^{6 \times 3}$; call this matrix \mathbf{V}_{123} . We can now see that the rows of $G_s(\mathbf{V}) = \sigma_g(\mathbf{W}_g s^+) \odot \mathbf{V}_{123}$ will approximate $\pm C c_i^\pm \mathbf{v}_i$, and the error bound in each coordinate is $Cb\epsilon$, since no coordinate of \mathbf{v}_i is greater than b . Then the vector $f(\mathbf{V}) = \mathbf{1}^T G_s(\mathbf{V})$ will approximate $\sum_{i=1}^3 C(c_i^+ - c_i^-) \mathbf{v}_i = \sum_{i=1}^3 c_i \mathbf{v}_i = F(\mathbf{V})$, and the error bound in each coordinate is at most the sum of the six row error bounds. \square