Received: 22 July 2023 | Revised: 5 January 2024 | Accepted: 1 February 2024

DOI: 10.1002/pro.4932

TOOLS FOR PROTEIN SCIENCE

THE PROTEIN SOCIETY WILEY

Protein structure accuracy estimation using geometry-complete perceptron networks

Alex Morehead | **Jian Liu** | **Jianlin Cheng**

Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, Missouri, USA

Correspondence
Alex Morehead, Jianlin Cheng, Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, Missouri, 65211, USA.
Email: acmwhb@missouri.edu; chengji@missouri.edu

Funding information
National Science Foundation, Grant/Award Numbers: DBI1759934, DBI2308699, IIS1763246; National Institutes of Health, Grant/Award Numbers: R01GM093123, R01GM146340; U.S. Department of Energy, Grant/Award Numbers: DE-AR0001213, DE-SC0020400, DE-SC0021303

Review Editor: Nir Ben-Tal

Abstract

Estimating the accuracy of protein structural models is a critical task in protein bioinformatics. The need for robust methods in the estimation of protein model accuracy (EMA) is prevalent in the field of protein structure prediction, where computationally-predicted structures need to be screened rapidly for the reliability of the positions predicted for each of their amino acid residues and their overall quality. Current methods proposed for EMA are either coupled tightly to existing protein structure prediction methods or evaluate protein structures without sufficiently leveraging the rich, geometric information available in such structures to guide accuracy estimation. In this work, we propose a geometric message passing neural network referred to as the geometry-complete perceptron network for protein structure EMA (GCPNet-EMA), where we demonstrate through rigorous computational benchmarks that GCPNet-EMA's accuracy estimations are 47% faster and more than 10% (6%) more correlated with ground-truth measures of per-residue (per-target) structural accuracy compared to baseline state-of-the-art methods for tertiary (multimer) structure EMA including AlphaFold 2. The source code and data for GCPNet-EMA are available on GitHub, and a public web server implementation is freely available.

KEYWORDS

3D graphs, accuracy estimation, deep learning, protein structure

1 | INTRODUCTION

Proteins are ubiquitous throughout the natural world, performing a plethora of crucial biological processes. Comprised of chains of amino acids, proteins carry out complex tasks throughout the bodies of living organisms, such as digestion, muscle growth, and hormone signaling. As a central notion in protein biology, the amino acid sequence of each protein uniquely determines its structure and, thereby, its function (Sadowski & Jones, 2009). However, the process of folding an amino acid sequence into a specific 3D protein structure has long been considered a fundamental challenge in protein biophysics (Dill & MacCallum, 2012).

Fortunately, in recent years, computational approaches to predicting the final state of protein folding (i.e., protein structure prediction) have advanced considerably (Jumper et al., 2021), to the degree that many have considered the problem of static protein tertiary structure prediction largely addressed (Al-Janabi, 2022). However, in relying on computational structure predictions for protein sequence inputs, a new problem in quality assessment arises (Kryshtafovych et al., 2019). In particular, how is one to estimate the accuracy of a predicted protein structure? Many computational approaches that aim to

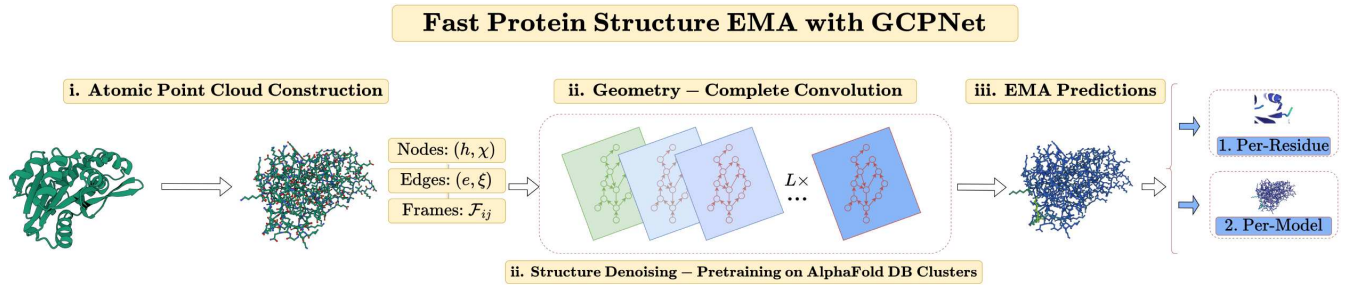

FIGURE 1 A high-level overview of GCPNet-EMA, our proposed method for protein structure EMA. Given an arbitrary 3D point cloud, GCPNet-EMA constructs a corresponding 3D graph representation of its input and learns latent scalar and vector features characterizing the input that can be used for downstream prediction tasks, following a structural denoising pretraining objective on AlphaFold Protein Structure Database cluster representatives corrupted with Gaussian noise. Accordingly, given a predicted 3D protein structure, GCPNet-EMA can provide both per-residue and per-model estimates of its structural accuracy. Zoom in for the best viewing experience.

answer this question have previously been proposed (e.g., Siew et al., 2000; Wallner & Elofsson, 2003; Shehu & Olson, 2010; Uziela et al., 2016; Cao et al., 2016; Olechnovič & Venclovas, 2017; Cheng et al., 2019; Maghrabi & McGuffin, 2020; Yang et al., 2020; Alshammari & He, 2020; Hiranuma et al., 2021; Baldassarre et al., 2021; McGuffin et al., 2021; Lensink et al., 2021; Akdel et al., 2022; Edmunds et al., 2023; Maghrabi et al., 2023). Nonetheless, previous methods for estimation of protein structural model accuracy (EMA) do not sufficiently utilize the rich, geometric information provided by 3D protein structure inputs directly as a methodological component, which suggests that future methods for EMA that can *learn* expressive geometric representations of 3D protein structures may provide an enhanced means by which to quickly and effectively estimate the accuracy of a predicted protein tertiary structure.

In this work, we introduce a geometric neural network, the geometry-complete perceptron network (GCPNet) for estimating the accuracy of 3D protein structures (called GCPNet-EMA). As illustrated in Figure 1, GCPNet-EMA receives as its primary network input a 3D point cloud, a representation naturally applicable to 3D protein structures when modeling these structures as graphs with nodes (i.e., residues represented by Ca atoms) positioned in 3D Euclidean space (Morehead et al., 2023). GCPNet-EMA then featurizes such 3D graph inputs as a combination of scalar and vector-valued features such as the type of a residue and the unit vector pointing from residue i to residue j, respectively. Subsequently, following pretraining on Gaussian noised-cluster representatives from the AlphaFold Protein Structure Database (Jumper et al., 2021; Varadi et al., 2021), GCPNet-EMA applies several layers of geometry-complete graph convolution (i.e., GCPConv) using a collection of node-specific and edge-specific geometry-complete perceptron (GCP) modules to learn an expressive scalar and vector-geometric representation of each of its 3D graph inputs (Morehead & Cheng, 2023a). Lastly, using its learned fine-tuning representations, GCPNet-EMA predicts a scalar structural accuracy value indicating the method's predicted lDDT score (Mariani et al., 2013) for each node (i.e., residue). Estimates of a protein structure's global (i.e., per-model) accuracy can then be calculated as the average of its residues' individual lDDT scores, following previous conventions for EMA (Chen et al., 2023).

2 | RESULTS AND DISCUSSIONS

For the following experiments, we adopt the tertiary (multimer) test datasets of Chen et al. (2023) Liu et al. (2023b) as introduced in these corresponding works for general tertiary, CASP15 multimer, and general multimer EMA, respectively. Additional details regarding the construction and composition of these three datasets are given in Section 4.2. Following Chen et al. (2023), for tertiary structure EMA, we report the same set of computational metrics to reflect each method's performance for EMA, including the mean squared error (MSE), mean absolute error (MAE), and Pearson's correlation coefficient (Cor) at a per-residue and per-model (target) level. Similarly, for multimer structure EMA, we report the per-target Pearson's correlation (Cor) and Spearman's correlation (SpearCor) of each multimer structure EMA method. The definitions for each of these metrics are as follows:

$$\text{MSE}: \frac{1}{n}\sum_{i=1}^{n}(y_i-\widehat{y}_i)^2, \quad \text{MAE}: \frac{1}{n}\sum_{i=1}^{n}|y_i-\widehat{y}_i|,$$

$$\text{Cor}: \frac{\sum_{i=1}^{n}\left(\widehat{y}_i-\bar{\widehat{y}}\right)(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}\left(\widehat{y}_i-\bar{\widehat{y}}\right)^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}},$$

$$\text{SpearCor}: 1-\frac{6\sum d_i^2}{n(n^2-1)},$$

TABLE 1 A comparison of GCPNet-EMA against baseline methods for protein tertiary structure EMA.

	Per-residue			Per-model		
Method	**MSE ↓**	**MAE ↓**	**Cor ↑**	**MSE ↓**	**MAE ↓**	**Cor ↑**
AF2-plDDT	0.0173	0.0888	0.6351	0.0105	0.0802	0.8376
DeepAccNet	0.0353	0.1359	0.3039	0.0249	0.1331	0.4966
VoroMQA	0.2031	0.4094	0.3566	0.1788	0.4071	0.3400
EnQA	0.0093	0.0723	0.6691	0.0031	0.0462	0.8984
EnQA-SE(3)	0.0102	0.0708	0.6224	0.0034	0.0434	0.8926
EnQA-MSA	0.0090	0.0653	0.6778	0.0027	0.0386	0.9001
GCPNet-EMA (pretraining, plDDT, and ESM)	0.0106	0.0724	0.7058	0.0031	0.0427	0.8687
GCPNet-EMA w/o pretraining	0.0107	0.0725	0.7048	0.0041	0.0482	0.8097
GCPNet-EMA w/ Null plDDT	0.6672	0.8022	0.2633	0.6305	0.7877	0.4131
GCPNet-EMA w/ null plDDT and w/o ESM	0.3342	0.5603	0.2139	0.3207	0.5548	0.2790
GCPNet-EMA w/o AF2 plDDT	0.0120	0.0759	0.6588	0.0051	0.0514	0.7633
GCPNet-EMA w/o pretraining or plDDT	0.0134	0.0803	0.6043	0.0066	0.0606	0.6744
GCPNet-EMA w/o ESM embeddings[a,b]	0.0092	**0.0648**	**0.7482**	0.0038	0.0420	0.8382
GCPNet-EMA w/o plDDT or ESM[a]	0.0105	0.0707	**0.7123**	0.0042	0.0461	0.8076

Note: Results for methods performing best are listed in bold, and results for methods performing second-best are underlined. Pretraining indicates that a method was pretrained on the 2.3 million tertiary structural cluster representatives of the AFDB (i.e., the afdb_rep_v4 dataset (Jamasb et al., 2024)) via a 3D residue structural denoising objective, in which small Gaussian noise is added to residue positions and a method is tasked with predicting the added noise.
Abbreviations: Cor, Pearson's correlation coefficient; MAE, mean absolute error; MSE, mean squared error.
[a]A method that was selected for deployment via our publicly available protein model quality assessment server.
[b]A method that is specialized for estimating the quality of AlphaFold-predicted protein structures.

where n is the number of samples across the test dataset; y_i is the ground-truth structural accuracy reported as a scalar value for the ith protein structure; $\widehat{y}_i$ is a method's predicted structural accuracy as a scalar value for the ith protein structure; $\overline{v}$ represents the mean of a variable v across the test dataset; and $d_i = \text{rank}(y_i - \widehat{y}_i)$ denotes the difference between the ranks of y_i and $\widehat{y}_i$ (w.r.t. the sets of ground-truth and predicted structural accuracy values, respectively).

As shown in Table 1, for tertiary structure EMA, GCPNet-EMA without ESM embedding inputs (Lin et al., 2023) outperforms all baseline methods (Jumper et al., 2021; Olechnovič & Venclovas, 2017; Hiranuma et al., 2021; Chen et al., 2023) in terms of its MAE and Pearson's correlation in predicting per-residue lDDT scores. Similarly, GCPNet-EMA without ESM embeddings achieves competitive per-residue MSE and per-model MAE values in predicting lDDT scores compared to EnQA-MSA (Chen et al., 2023), the most recent state-of-the-art method for protein structure EMA. Analyzed jointly, GCPNet-EMA offers state-of-the-art lDDT predictions for each residue in a predicted protein structure and competitive per-model predictions overall, with more than 10% greater correlation to ground-truth lDDT scores for each residue compared to EnQA-MSA. Notably, in doing so, GCPNet-EMA also outperforms the lDDT score estimations produced by AlphaFold 2 (Jumper et al., 2021) in the form of its predicted lDDT (plDDT) scores. These results suggest that GCPNet-EMA should be broadly applicable for a variety of tasks in protein bioinformatics related to local and global tertiary structure EMA. In Figure 2, we show an example of a protein in the tertiary structure EMA test dataset for which AlphaFold overestimates the accuracy of its predicted structure but for which GCPNet-EMA's plDDT scores are quantitatively and qualitatively much closer to the ground-truth lDDT values, likely due to its large-scale structure-denoising-based pretraining on the afdb_rep_v4 dataset (Jamasb et al., 2024), a redundancy-reduced label-free subset of the AlphaFold Protein Structure Database (AFDB) (Jumper et al., 2021; Varadi et al., 2021).

Concerning CASP15 multimer structure EMA, Table 2 shows that GCPNet-EMA provides the most balanced performance compared to four single-model baseline methods (Jumper et al., 2021; Chen et al., 2023, 2023; Olechnovič & Venclovas, 2023) in terms of its per-target Pearson's and Spearman's correlation as well as its performance for ranking loss, for which it is better than AlphaFold 2 (i.e., AlphaFold-Multimer for multimeric benchmarking) plDDT yet marginally outperformed by VoroMQA-dark which is mostly uncorrelated with the quality of an individual decoy. Note that in contrast to

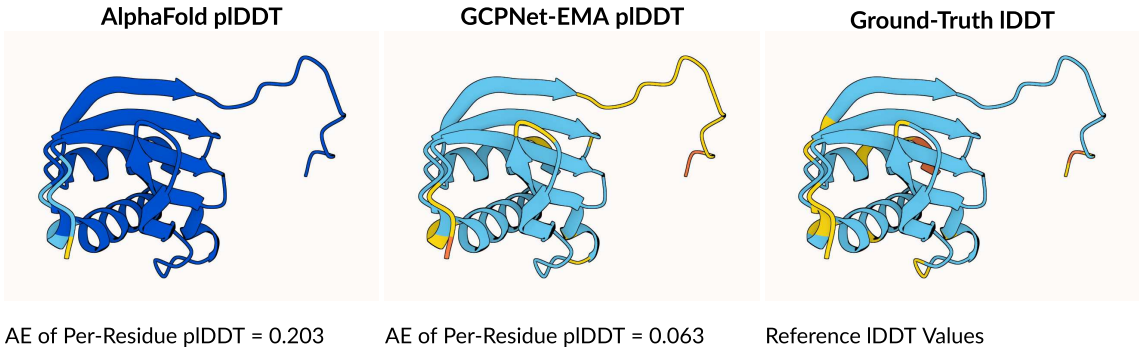

FIGURE 2 An example of an AlphaFold-predicted test protein (PDB ID: 6W77, chain K) for which AlphaFold assigns overly-optimistic "very high" confidence values for its structural accuracy, whereas GCPNet correctly assigns "high" confidence values to the structure. Note in all the above subfigures that the coloring scheme for the per-residue plDDT values (i.e., structural confidence values) follows that used throughout the AlphaFold Protein Structure Database (Jumper et al., 2021; Varadi et al., 2021), where "very high" plDDT corresponds to blue; "high" plDDT corresponds to cyan; "low" plDDT corresponds to yellow; and "very low" plDDT corresponds to orange. As subcaptions for this figure, we also report the absolute error (AE) of each method's per-residue plDDT, averaged across all residues in the protein chain, to quantify each method's EMA performance. Zoom in for the best viewing experience.

TABLE 2 A comparison of GCPNet-EMA against baseline methods for CASP15 protein multimer structure EMA.

Method	Cor	SpearCor	Loss
AF2-plDDT	**0.3402**	**0.2641**	0.1106
DProQA	0.0795	0.0545	0.1199
EnQA-MSA	0.2550	0.2378	0.1036
VoroIF-GNN-score	0.0639	0.0873	0.1342
Average-VoroIF-GNN-residue-pCAD-score	−0.0156	−0.0326	0.1499
VoroMQA-dark	−0.0872	−0.0119	**0.0860**
GCPNet-EMA	<u>0.3056</u>	<u>0.2567</u>	<u>0.0970</u>
GCPNet-EMA w/o ESM embeddings	0.2592	0.1969	0.1292
GCPNet-EMA w/o plDDT	0.0853	0.0450	0.1337

Note: Results for methods performing best are listed in bold, and results for methods performing second-best are underlined. Note that all versions of GCPNet-EMA benchmarked for multimer structure EMA were pretrained using the AFDB, using the same structural denoising objective investigated in Table 1.
Abbreviations: Cor, Pearson's correlation coefficient; Loss, ranking loss defined as the target-averaged difference between the TM-score of a method's top-ranked decoy structure and that of the ground-truth top-ranked decoy structure for all decoys corresponding to a given target; SpearCor, Spearman's rank correlation coefficient.

tertiary structure EMA, for multimer structure EMA, we instead assess a method's ability to predict (a quantity correlated with) the TM-score of a given decoy corresponding to a protein target. Overall, these results demonstrate that, compared to state-of-the-art single-model multimer EMA methods, GCPNet-EMA offers robust, balanced multimer EMA performance in contemporary real-world EMA benchmarks such as CASP15.

For general PDB multimer structure EMA, Table 3 shows that GCPNet-EMA outperforms 4 single-model baseline methods (Jumper et al., 2021; Chen et al., 2023, 2023; Olechnovič & Venclovas, 2023) in terms of its per-target Pearson's and Spearman's correlation as well as its state of the art performance for ranking loss, for which it is tied only with AlphaFold-Multimer plDDT. Notably, without plDDT as an input feature, GCPNet-EMA still surpasses the Pearson's and Spearman's correlation of DProQA, a recent state-of-the-art method for protein multimer structure EMA. Overall, GCPNet-EMA offers 6% greater Spearman's correlation to ground-truth TM-scores for each decoy of a given multimer target compared to AlphaFold-Multimer, the second-best-performing method. Observing that GCPNet-EMA is successfully able to generalize from being trained for tertiary structure EMA to being evaluated for multimer structure EMA, these results suggest that GCPNet-EMA should be useful for a variety of tasks related to accuracy estimation of multimeric structures.

TABLE 3 A comparison of GCPNet-EMA against baseline methods for PDB protein multimer structure EMA.

Method	Cor	SpearCor	Loss
AF2-plDDT	0.3654	0.2799	0.0563
DProQA	0.1403	0.1563	0.0816
EnQA-MSA	0.3303	0.2395	0.0577
VoroIF-GNN-score	0.1017	0.1213	0.0715
Average-VoroIF-GNN-residue-pCAD-score	0.0483	0.0355	0.1198
VoroMQA-dark	0.0099	0.1036	0.0835
GCPNet-EMA	**0.3756**	**0.2971**	**0.0563**
GCPNet-EMA w/o ESM embeddings	0.2920	0.2387	0.0799
GCPNet-EMA w/o plDDT	0.2176	0.1973	0.1082

Note: Results for methods performing best are listed in bold, and results for methods performing second-best are underlined. Note that all versions of GCPNet-EMA benchmarked for multimer structure EMA were pretrained using the AFDB, using the same structural denoising objective investigated in Table 1. Abbreviations: Cor, Pearson's correlation coefficient; Loss, ranking loss defined as the target-averaged difference between the TM-score of a method's top-ranked decoy structure and that of the ground-truth top-ranked decoy structure for all decoys corresponding to a given target; SpearCor, Spearman's rank correlation coefficient.

TABLE 4 A comparison of the runtime of GCPNet-EMA against the runtime of EnQA-MSA, using all 56 tertiary structure EMA test decoys as each model's inputs.

Method	Average prediction speed ↓
EnQA-MSA	15.3 s
GCPNet-EMA	**8.1** s

Note: Results for the fastest method are listed in bold.

Lastly, in Table 4, we compare the runtime of GCPNet-EMA to the runtime of EnQA-MSA using the 56 decoys comprising the tertiary structure EMA test dataset referenced in Table 1. The results here show that GCPNet-EMA offers 47% faster EMA predictions for arbitrary protein structure inputs compared to EnQA-MSA, highlighting real-world utility in incorporating GCPNet-EMA into modern protein structure prediction pipelines.

3 | CONCLUSIONS

In this work, we introduced GCPNet-EMA for fast protein structure EMA. Our experimental results demonstrate that GCPNet-EMA offers state-of-the-art (competitive) estimation performance for per-residue (per-model) tertiary structural accuracy measures such as plDDT, while offering fast prediction runtimes within a publicly-available web server interface. Moreover, GCPNet-EMA achieves state-of-the-art PDB multimer structure EMA performance across all metrics and performs competitively for CASP15 multimer EMA. Consequently, as an open-source software utility, GCPNet-EMA should be widely applicable within the field of protein bioinformatics for understanding the relationship between predicted protein structures and their native structure counterparts. In future work, we believe it would be worthwhile to explore applications of GCPNet-EMA's predictions of protein structure accuracy to better understand the presence (or absence) of disordered regions in protein structures, to better characterize the potential protein dynamics in effect.

4 | MATERIALS AND METHODS

In this section, we will describe our proposed method, GCPNet-EMA, in greater detail to better understand how it can learn geometric representations of protein structure inputs for downstream tasks.

Towards this end, we introduce our geometric neural network architecture which we refer to as the geometry-complete SE(3)-equivariant perceptron network (GCPNet). We illustrate the GCPNet algorithm in Figure 1 and outline it in Algorithm 1. Subsequently, we expand on our definition for **GCP** and **GCPConv** in Sections 4.1 and 4.2.1, respectively. As shown by Morehead and Cheng (2023a), by construction GCPNets possess the following three properties, which as we will discuss will be useful for predicting protein structure accuracy measurements.

1. **Property:** GCPNets *are SE(3)-equivariant, in that they preserve 3D transformations acting upon their vector inputs.*
2. **Property:** GCPNets *are geometry self-consistent, in that they preserve rotation invariance for their scalar features.*

3. **Property:** GCPNets *are geometry-complete, in that they encode direction-robust local reference frames for each node.*

4.1 | The geometry-complete perceptron module

GCPNet, as illustrated in Figure 1 and shown in Algorithm 1, represents the features for nodes within its protein graph inputs as a tuple (h,χ) to learn scalar features $\left(h \in \mathbb{R}^h\right)$ jointly with vector-valued features $(\chi \in \mathbb{R}^{m\times 3})$. Likewise, GCPNet represents the features for edges in its protein graph inputs as a tuple (e,ξ) to learn scalar features $(e \in \mathbb{R}^e)$ jointly with vector-valued features $(\xi \in \mathbb{R}^{x\times 3})$. Hereon, to be concise, we refer to both node and edge feature tuples as (s,V). Lastly, GCPNet denotes each node's position in 3D space as a dedicated, translation-equivariant vector feature $x \in \mathbb{R}^{1\times 3}$.

Defining notation for the GCP *module.* Let λ represent an integer downscaling hyperparameter (e.g., 3), and let $\mathcal{F}_{ij} \in \mathbb{R}^{3\times 3}$ denote the SO(3)-equivariant (i.e., 3D rotation-equivariant) frames constructed using the **Localize** operation in Algorithm 1, as previously described by Morehead and Cheng (2023a). We then use the local frames $\mathcal{F}_{ij}$ to define the **GCP** encoding process for 3D graph inputs. Specifically, for an optional time index t, we define these frame encodings as $\mathcal{F}_{ij}^t = \left(a_{ij}^t, b_{ij}^t, c_{ij}^t\right)$, with $a_{ij}^t = \frac{x_i^t - x_j^t}{\|x_i^t - x_j^t\|}$, $b_{ij}^t = \frac{x_i^t \times x_j^t}{\|x_i^t \times x_j^t\|}$, and $c_{ij}^t = a_{ij}^t \times b_{ij}^t$, respectively. Notably, Morehead and Cheng (2023a, 2023b) show how these frame encodings allow networks that incorporate them to effectively detect and leverage for downstream tasks the potential effects of molecular chirality on protein structure.

Using V *to express vector and frame representations within each* GCP *module.* After initially projecting vector-valued input features ($\chi \in \mathbb{R}^{m\times 3}$ and $\xi \in \mathbb{R}^{x\times 3}$) to each be of hidden dimensionality $\mathbb{R}^{r\times 3}$ and have their coordinates axes permuted to $\mathbb{R}^{3\times r}$, the **GCP** module separately expresses vector representations V for nodes (edges) and local frames, the former of which is to have its representations downscaled by a factor of λ, using the following two equations, respectively.

$$z = \left\{ v\boldsymbol{w}_{d_z} | \boldsymbol{w}_{d_z} \in \mathbb{R}^{r\times(r/\lambda)} \right\}, \tag{1}$$

$$V_s = \left\{ v\boldsymbol{w}_{d_s} | \boldsymbol{w}_{d_s} \in \mathbb{R}^{r\times(3\times 3)} \right\}. \tag{2}$$

Expressing s' *as scalar representations for each* GCP *module.* To express scalar representations, the **GCP** module computes two invariant sources of information from V and combines them with s, as follows:

$$q_{ij} = \left(V_s \cdot \mathcal{F}_{ij}\right) \in \mathbb{R}^9, \tag{3}$$

$$q = \begin{cases} \frac{1}{|\mathcal{N}(i)|} \sum_{j\in\mathcal{N}(i)} q_{ij} & \text{if } V_s \text{ represents nodes} \\ q_{ij} & \text{if } V_s \text{ represents edges} \end{cases}, \tag{4}$$

$$s_{(s,q,z)} = s \cup q \cup \|z\|_2, \tag{5}$$

where $\mathcal{N}(\cdot)$ denotes the neighbors of a node; where $\cdot$ represents the inner product; and where $\|\cdot\|_2$ indicates the L_2 norm of a vector. Subsequently, let $s_{(s,q,z)} \in \mathbb{R}^{t+9+(r/\lambda)}$ with hidden dimensionality $(t+9+(r/\lambda))$ be projected to s' with hidden dimensionality t', with t denoting the hidden dimensionality of s:

$$s_v = \left\{ s_{(s,q,z)}\boldsymbol{w}_s + \boldsymbol{b}_s | \boldsymbol{w}_s \in \mathbb{R}^{(t+9+(r/\lambda))\times t'} \right\}, \tag{6}$$

$$s' = \sigma_s(s_v). \tag{7}$$

Computing V' *as updated vector representations within each* GCP *module.* The **GCP** module lastly updates vector representations using the following equations:

$$V_u = \left\{ z\boldsymbol{w}_{u_z} | \boldsymbol{w}_{u_z} \in \mathbb{R}^{(r/\lambda)\times r'} \right\}, \tag{8}$$

$$V' = \left\{ V_u \bigodot \sigma_g\left(\sigma^+(s_v)\boldsymbol{w}_g + \boldsymbol{b}_g\right) | \boldsymbol{w}_g \in \mathbb{R}^{t'\times r'} \right\}, \tag{9}$$

where $\bigodot$ represents element-wise multiplication and the gating operation σ_g is applied row-wise to V' to preserve SO(3) equivariance for vector features.

ALGORITHM 1 GCPNet for estimation of protein structure model accuracy

1: **Input:** $(h_i \in \mathbf{H}, \chi_i \in \boldsymbol{\chi})$, $\left(e_{ij} \in \mathbf{E}, \xi_{ij} \in \boldsymbol{\xi}\right)$, $x_i \in \mathbf{X}$, graph $\mathcal{G}$
2: Initialize $\mathbf{X}^0 = \mathbf{X}^C \leftarrow$ **Centralize**$(\mathbf{X})$
3: $\mathcal{F}_{ij} =$ **Localize**$\left(x_i \in \mathbf{X}^0, x_j \in \mathbf{X}^0\right)$
4: Project $\left(h_i^0, \chi_i^0\right)$, $\left(e_{ij}^0, \xi_{ij}^0\right) \leftarrow$ **GCP**$_e\left((h_i, \chi_i), (e_{ij}, \xi_{ij}), \mathcal{F}_{ij}\right)$
5: **for** $l = 1$ **to** L **do**
6: $\left(h_i^l, \chi_i^l\right), =$ **GCPConv**$^l\left(\left(h_i^{l-1}, \chi_i^{l-1}\right), \left(e_{ij}^0, \xi_{ij}^0\right), \mathcal{F}_{ij}\right)$
7: **end for**
8: Project $h_i^L, \leftarrow$ **GCP**$_p\left(\left(h_i^l, \chi_i^l\right), \left(e_{ij}^0, \xi_{ij}^0\right), \mathcal{F}_{ij}\right)$
9: **Output:** h_i^L

To summarize, the **GCP** module learns tuples (s, V) of scalar and vector features a total of ω times to derive rich scalar and vector-valued features. The module does so by blending both feature types iteratively with the local geometric information provided by the chirality-sensitive frame encodings $\mathscr{F}_{ij}$.

4.2 | Learning from 3D protein graphs with GCPNet

In this section, we will describe how the **GCP** module can be used to perform 3D graph convolution with protein graph inputs, as illustrated in Algorithm 1.

4.2.1 | A geometry-complete graph convolution layer

GCPNet defines a single layer l of geometry-complete graph convolution (**GCPConv**) as

$$n_i^l = \phi^l\left(n_i^{l-1},\ \mathcal{A}_{\forall j \in \mathcal{N}(i)}\Omega_\omega^l\left(n_i^{l-1}, n_j^{l-1}, e_{ij}, \mathscr{F}_{ij}\right)\right), \tag{10}$$

where $n_i^l = \left(h_i^l, \chi_i^l\right)$; $e_{ij} = \left(e_{ij}^0, \xi_{ij}^0\right)$; $\mathcal{N}(i)$ represents the neighbors of node n_i, selected using a distance-based metric such as k-nearest neighbors or a radial distance cutoff; l signifies the hidden dimensionality of the network; $\mathcal{A}$ is a permutation-invariant aggregation function; and Ω_ω represents a message-passing function corresponding to the ωth **GCP** message-passing layer.

At the start of each graph convolution layer, messages between source nodes i and neighboring nodes j are comprised as

$$m_{ij}^0 = \mathbf{GCP}\left(n_i^0 \cup n_j^0 \cup e_{ij}, \mathscr{F}_{ij}\right), \tag{11}$$

where $\cup$ represents concatenation. Up to the ωth iteration, each message is updated by the m-th message update layer residually as

$$\Omega_\omega^l = \mathbf{ResGCP}_\omega^l\left(m_{ij}^{l-1}, \mathscr{F}_{ij}\right), \tag{12}$$

$$\mathbf{ResGCP}_\eta^l\left(z_i^{l-1},\ \mathscr{F}_{ij}\right) = z_i^{l-1} + \mathbf{GCP}_\eta^l\left(z_i^{l-1},\ \mathscr{F}_{ij}\right). \tag{13}$$

Updated node features $\widehat{n}^l$ are derived residually using an aggregation of generated messages as

$$\widehat{n}^l = n^{l-1} \cup f\left(\left\{\left(g_{e^\omega, v_i}^l \Omega_{e^\omega, v_i}^l,\ \ \Omega_{\xi^\omega, v_i}^l\right) | v_i \in \mathcal{V}\right\}\right), \tag{14}$$

where f represents a summation or a mean function that is invariant to node order permutations; Ω_{e^ω, v_i}^l denotes scalar message features collected for node i; $\Omega_{\xi^\omega, v_i}^l$ represents vector message features associated with node i; and $g_{e^\omega}^l$ represents the binary-valued (i.e., [0, 1]) output of a scalar message attention (gating) function, expressed as $g_{e^\omega}^l = \sigma_{inf}\left(\phi_{inf}^l\left(\Omega_{e^\omega}^l\right)\right)$ with $\phi_{inf} : \mathbb{R}^e \to [0,1]^1$ mapping from high-dimensional scalar edge feature space to a single dimension and σ denoting a sigmoid activation function.

Each graph convolution layer then employs a node-specific feed-forward network to update node representations. In particular, a linear **GCP** function with shared weights ϕ_f is applied to $\widehat{n}^l$, followed by r **GCP** modules. Such operations are concisely portrayed:

$$\widetilde{n}_{r-1}^l = \phi_f^l\left(\widehat{n}^l\right), \tag{15}$$

$$n^l = \mathbf{GCP}_r^l\left(\widetilde{n}_{r-1}^l\right). \tag{16}$$

4.2.2 | Designing GCPNet for estimation of protein structure model accuracy

In this remaining section, we discuss GCPNet-EMA—the overall GCPNet-based protein structure EMA algorithm (Algorithm 1).

Line 2 of Algorithm 1 uses the **Centralize** operation to remove the center of mass from each node (atom) position in a protein graph input to ensure that such positions are 3D translation-invariant for the remainder of the algorithm's execution.

Subsequently, the **Localize** operation on Line 3 crafts translation-invariant and SO(3)-equivariant frame encodings $\mathscr{F}_{ij}^t = \left(a_{ij}^t, b_{ij}^t, c_{ij}^t\right)$. As described in more detail in Morehead and Cheng (2023a), these frame encodings are chirality-sensitive and direction-robust for edges, imbuing networks that incorporate them with the ability to more easily detect the influence of molecular chirality on protein structure.

Notably, Line 4 uses $\mathbf{GCP}_e$ to initially embed our node and edge feature inputs into scalar and vector-valued values, respectively, using encodings of geometric frames. Thereafter, Lines 5–7 show how each layer of graph convolution is applied iteratively via $\mathbf{GCPConv}^l$, starting from these initial node and edge feature embeddings. Important to note is that information flow originating from the geometric frames $\mathscr{F}_{ij}$ is always maintained to simplify the network's synthesis of

TABLE 5 Features used by the GCPNet-EMA models with a k-NN ($k = 16$) Cα protein graph representation.

Type	Symmetry	Feature name
Node	Invariant	Residue type
Node	Invariant	Positional encoding
Node	Invariant	Virtual dihedral and bond angles over the Cα trace
Node	Invariant	Residue backbone dihedral angles
Node	Invariant	(Optional) residue-wise ESM embeddings
Node	Invariant	(Optional) residue-wise AlphaFold 2 plDDT
Node	Equivariant	Residue-sequential forward and backward (orientation) vectors
Edge	Invariant	Euclidean distance between connected Cα atoms
Edge	Equivariant	Directional vector between connected Cα atoms

TABLE 6 Specifications of each GCPNet-EMA model, where the learning rate and weight decay rate (for finetuning only) were determined by a grid search targeting the EMA dataset's validation split.

Specification	Value
Number of GCP layers	6
Number of GCPConv operations per GCP layer	4
Hidden dimensionality of each GCP/GCPConv layer's embeddings	128
Optimizer	Adam
Learning rate	$1e^{-5}$
Weight decay rate	$1e^{-5}$
Batch size	16
Number of trainable parameters	4 M
Pretraining runtime on afdb_rep_v4 (using 4 80GB NVIDIA GPUs)	5 days
Finetuning runtime on the tertiary structure EMA dataset (using one 24GB NVIDIA GPU)	8 h

information derived from its geometric local frames in each layer.

Lines 8 through 9 finalize the GCPNet-EMA algorithm for EMA by performing feature projections via $\mathbf{GCP}_p$ to conclude the forward pass of GCPNet by returning its final node-specific scalar outputs.

4.2.3 | Network outputs

To summarize, GCPNet-EMA receives a 3D graph input $\mathcal{G}$ with node positions $\boldsymbol{x}$, scalar node and edge features, h and e, as well as vector-valued node and edge features, χ and ξ, where all of such features used are listed in Table 5. GCPNet then predicts scalar node-level properties while maintaining SE(3) invariance to estimate the per-residue and per-model accuracy of a given protein structure, to avoid imposing an arbitrary 3D reference frame on the model's final prediction.

4.2.4 | Training, evaluating, and optimizing the network

As referenced in Table 6, we trained each GCPNet-EMA model on the tertiary structure EMA cross-validation dataset as discussed in Section 2, using its 80%–20% training and validation data splits for training and validation, respectively. Subsequently, for finetuning the afdb_rep_v4-pretrained GCPNet model weights, we performed a grid search for the best hyperparameters to optimize a model's performance on the EMA validation dataset, searching for the network's best combination of learning rate and weight decay rate within the intervals of $[1e^{-5}, 4e^{-5}, 3e^{-4}, 1e^{-3}]$ and $[1e^{-5}, 1e^{-4}, 1e^{-3}]$, respectively. The epoch checkpoint that yielded a model's best lDDT L1 loss value on the tertiary structure EMA validation dataset was then tested on the tertiary and multimer structure EMA test datasets as described in Section 2 for fair comparisons with prior methods. Note that we used the same training and evaluation procedure as well as hyperparameters in our ablation experiments with GCPNet-EMA. Moreover, pretraining was performed using Gaussian-noised afdb_rep_v4 structures with noised residue atom coordinates $\widetilde{x}_i$ defined as $\widetilde{x}_i = x_i + \sigma\varepsilon_i$, where $\sigma = 0.1$and $\varepsilon_i \sim \mathcal{N}(0, I_3)$. Notably, as shown by Zaidi et al. (2023), this corresponds to approximating the Boltzmann distribution with a mixture of Gaussian distributions.

4.3 | Datasets

To evaluate the effectiveness of our proposed GCPNet-based EMA method (i.e., GCPNet-EMA) compared to baseline state-of-the-art methods for EMA, we adopted the experimental configuration of Chen et al. (2023). This configuration includes a standardized tertiary structure EMA cross-validation dataset for the training, validation, and testing of machine learning models, a dataset that we make publicly available at https://zenodo.org/record/8150859. As described by Chen et al. (2023), this cross-validation dataset is comprised of 4940 decoys

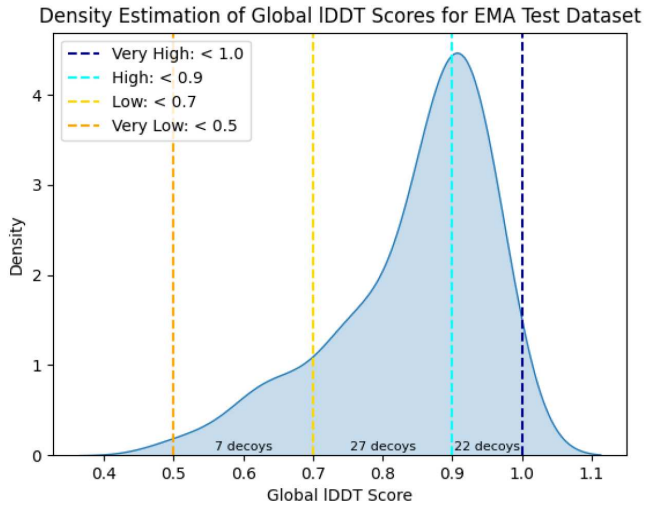

FIGURE 3 The distribution of global plDDT scores for each decoy in the tertiary structure EMA test dataset. This dataset, comprised of 56 decoys for 49 targets, consists of 22 very high quality decoys, 27 high quality decoys, and 7 low quality decoys, thereby closely resembling the data distributions used in similar benchmarks such as CAMEO.

(3906 targets) for training, 1236 decoys (1166 targets) for validation, and 56 decoys (49 targets) for testing, where such data splits are constructed such that no decoy (target) within the training or validation dataset belongs to the same SCOP family (Andreeva et al., 2014) as any decoy within the test dataset. Decoy structures were generated for each corresponding protein target using AlphaFold 2 for structure prediction (Jumper et al., 2021). We evaluate each method on the same 56 decoys (49 targets) contained in the test dataset to ensure a fair comparison between each method. Such test decoys, as illustrated in Figure 3, are predominantly ranked as "high" and "very high" quality decoys (i.e., lDDT values falling in the ranges of [0.7, 0.9] and [0.9, 1.0], respectively (Jumper et al., 2021; Varadi et al., 2021)), with the seven remaining decoys being of "low" structural accuracy as determined by having an lDDT value in the range of [0.5, 0.7]. We argue that evaluating methods in such a test setting is reasonable given that (1) the Continuous Automated Model EvaluatiOn (CAMEO) quality assessment category (Robin et al., 2021) employs a decoy quality distribution similar to that of the EMA test dataset; and (2) most protein structural decoys generated today are produced using high-accuracy methods such as AlphaFold 2.

Additionally, to rigorously evaluate the generalization capability and performance of GCPNet-EMA in the context of multimer structure EMA, we adopted a benchmark dataset of 100 hetero-multimer protein complexes from PDB entries released after AlphaFold-Multimer (i.e., between April 1, 2022 and December 9, 2022), which was previously compiled by (Liu et al., 2023b). Selected complexes, for each of which 350 decoy structures were generated by feeding 14 kinds of MSA_{paired} in MULTICOM to AlphaFold-Multimer without any templates information, were meticulously filtered to ensure quality and non-redundancy via the following criteria.

1. Sequence length: $<1{,}536$ residues.
2. Resolution: <4Å.
3. Number of chains: <8.
4. Hetero-multimer definition: Sequence identity between chains <0.9.
5. Inter-chain contacts: At least 10 inter-chain residue-residue pairs with a minimum heavy atom distance of <5Å.
6. Sequence similarity to known structures: <0.4 sequence identity with monomer chains in the PDB prior to April 1, 2022 and no significant template hits (e.g., *e*-value >1) in the MULTICOM monomer template database (Liu et al., 2023a).
7. Redundancy reduction: Clustering of subunits using MMseqs2 with a 0.3 sequence identity threshold and assigning the cluster ID of the hetero-multimer by the combination of the clusters of the subunits, followed by selection of the highest-resolution structure from each cluster ID of the hetero-multimers.

This general PDB multimer EMA dataset, characterized by its stringent filtering and focus on recently released hetero-multimers, provides a valuable benchmark for assessing the performance of multimer structure EMA methods, particularly in the context of challenging hetero-multimeric complexes. Furthermore, by way of its construction, it minimizes potential overlap between the tertiary structure EMA training and testing dataset, allowing for a meaningful assessment of each method's performance for multimer structure EMA. Note that the average TM-score of a decoy structure in this dataset is 0.7522, which as one might expect is slightly lower than that of the tertiary structure EMA dataset.

In conjunction with the PDB multimer EMA dataset, to compile a CASP15 multimer EMA test dataset we collected decoy structures generated by MULTICOM for the CASP15 assembly targets (Liu et al., 2023b). Note that 10 assembly targets (i.e., H1111, H1114, H1135, H1137, H1171, H1172, H1185, T1115o, T1176o, and T1192o) are not included due to various factors such as computational resource limitations and unavailable native structures or the presence of multiple conformations in native structures. As a result, this CASP15 MULTICOM multimer EMA dataset is comprised of an average of 254 decoy structures per target, all generated by AlphaFold-Multimer, across 31 assembly targets.

AUTHOR CONTRIBUTIONS

AM and JC conceived the project. AM designed the experiments. AM developed the source code. AM performed the primary experiments and data collection for tertiary structure quality assessment, and JL performed the experiments and data collection for multimer structure quality assessment. AM and JL analyzed the data. JC acquired the funding. AM, JL, and JC wrote the manuscript. AM, JL, and JC reviewed and edited the manuscript.

ACKNOWLEDGMENTS

This work was supported by one U.S. NSF grant (DBI2308699) and two U.S. NIH grants (R01GM093123 and R01GM146340).

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

DATA AVAILABILITY STATEMENT

The source code, data, and instructions to train GCPNet-EMA, reproduce our results, or estimate the accuracy of predicted protein structures are freely available at https://github.com/BioinfoMachineLearning/GCPNet-EMA, and a public web server implementation is freely available at http://gcpnet-ema.missouri.edu.

ORCID

Alex Morehead https://orcid.org/0000-0002-0586-6191
Jian Liu https://orcid.org/0000-0002-7570-8690
Jianlin Cheng https://orcid.org/0000-0003-0305-2853

REFERENCES

Akdel M, Pires DE, Pardo EP, Jänes J, Zalevsky AO, Mészáros B, et al. A structural biology community assessment of AlphaFold2 applications. Nat Struct Mol Biol. 2022;29(11):1056–67.

Al-Janabi A. Has DeepMind's AlphaFold solved the protein folding problem? Fut Sci. 2022;72:73–6.

Alshammari M, He J. Combine Cryo-EM density map and residue contact for protein structure prediction: a case study. Proceedings of the 11th ACM international conference on bioinformatics, computational biology and health informatics. Association for Computing Machinery, New York, USA; 2020. p. 1–6. https://doi.org/10.1145/3388440.3414708

Andreeva A, Howorth D, Chothia C, Kulesha E, Murzin AG. SCOP2 prototype: a new approach to protein structure mining. Nucleic Acids Res. 2014;42(D1):D310–4.

Baldassarre F, Menéndez Hurtado D, Elofsson A, Azizpour H. GraphQA: protein model quality assessment using graph convolutional networks. Bioinformatics. 2021;37(3):360–6.

Cao R, Bhattacharya D, Hou J, Cheng J. DeepQA: improving the estimation of single protein model quality with deep belief networks. BMC Bioinform. 2016;17:1–9.

Chen C, Chen X, Morehead A, Wu T, Cheng J. 3D-equivariant graph neural networks for protein model quality assessment. Bioinformatics. 2023;39(1):btad030.

Chen X, Morehead A, Liu J, Cheng J. A gated graph transformer for protein complex structure quality assessment and its performance in CASP15. Bioinformatics. 2023;39(Suppl_1):i308–17.

Cheng J, Choe MH, Elofsson A, Han KS, Hou J, Maghrabi AH, et al. Estimation of model accuracy in CASP13. Proteins: Struct Funct Bioinform. 2019;87(12):1361–77.

Dill KA, MacCallum JL. The protein-folding problem, 50 years on. Science. 2012;338(6110):1042–6.

Edmunds NS, Alharbi SM, Genc AG, Adiyaman R, McGuffin LJ. Estimation of model accuracy in CASP15 using the M odFOLDdock server. Proteins: Struct Funct Bioinform. 2023;91:1871–8.

Hiranuma N, Park H, Baek M, Anishchenko I, Dauparas J, Baker D. Improved protein structure refinement guided by deep learning based accuracy estimation. Nat Commun. 2021;12(1):1340.

Jamasb AR, Morehead A, Joshi CK, Zhang Z, Didi K, Mathis SV, et al. Evaluating representation learning on the protein structure universe. The twelfth international conference on learning representations; 2024. p. 14. https://openreview.net/forum?id=sTYuRVrdK3

Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, et al. Highly accurate protein structure prediction with AlphaFold. Nature. 2021;596(7873):583–9.

Kryshtafovych A, Schwede T, Topf M, Fidelis K, Moult J. Critical assessment of methods of protein structure prediction (CASP)–round XIII. Proteins: Struct Funct Bioinform. 2019;87(12):1011–20.

Lensink MF, Brysbaert G, Mauri T, Nadzirin N, Velankar S, Chaleil RA, et al. Prediction of protein assemblies, the next frontier: the CASP14-CAPRI experiment. Proteins: Struct Funct Bioinform. 2021;89(12):1800–23.

Lin Z, Akin H, Rao R, Hie B, Zhu Z, Lu W, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. Science. 2023;379(6637):1123–30.

Liu J, Guo Z, Wu T, Roy RS, Chen C, Cheng J. Improving AlphaFold2-based protein tertiary structure prediction with MULTICOM in CASP15. Commun Chem. 2023a;6(1):188.

Liu J, Guo Z, Wu T, Roy RS, Quadir F, Chen C, et al. Enhancing alphafold-multimer-based protein complex structure prediction with MULTICOM in CASP15. Commun Biol. 2023b;6(1):1140.

Maghrabi AH, Aldowsari FM, McGuffin LJ. Quality estimates for 3D protein models. Homology modeling: methods and protocols. Springer, New York, USA; 2023. p. 101–18. https://doi.org/10.1007/978-1-0716-2974-1_6

Maghrabi AH, McGuffin LJ. Estimating the quality of 3D protein models using the ModFOLD7 server. Protein Struct Pred. 2020;2165:69–81.

Mariani V, Biasini M, Barbato A, Schwede T. lDDT: a local superposition-free score for comparing protein structures and models using distance difference tests. Bioinformatics. 2013;29(21):2722–8.

McGuffin LJ, Aldowsari FM, Alharbi SM, Adiyaman R. ModFOLD8: accurate global and local quality estimates for 3D protein models. Nucleic Acids Res. 2021;49(W1):W425–30.

Morehead A, Cheng J. Geometry-complete perceptron networks for 3D molecular graphs. AAAI deep learning on graphs. Oxford University Press; Volume 2023; 2023a. p. 8.

Morehead A, Cheng J. Geometry-complete diffusion for 3D molecule generation. ICLR 2023 – machine learning for drug discovery workshop; 2023b. p. 1–15. https://openreview.net/forum?id=X-tLu3OUE-d

Morehead A, Ruffolo JA, Bhatnagar A, Madani A. Towards joint sequence-structure generation of nucleic acid and protein complexes with SE(3)-discrete diffusion. NeurIPS 2023 workshop on machine learning in structural biology. Neural Information Processing Systems, San Diego, CA; 2023. p. 14.

Olechnovič K, Venclovas Č. VoroMQA: assessment of protein structure quality using interatomic contact areas. Proteins: Struct Funct Bioinform. 2017;85(6):1131–45.

Olechnovič K, Venclovas Č. VoroIF-GNN: Voronoi tessellation-derived protein-protein interface assessment using a graph neural network. Proteins: Struct Funct Bioinform. 2023;91(12):1879–88.

Robin X, Haas J, Gumienny R, Smolinski A, Tauriello G, Schwede T. Continuous automated model evaluation (CAMEO)–perspectives on the future of fully automated evaluation of structure prediction methods. Proteins: Struct Funct Bioinform. 2021;89(12):1977–86.

Sadowski M, Jones D. The sequence–structure relationship and protein function prediction. Curr Opin Struct Biol. 2009;19(3):357–62.

Shehu A, Olson B. Guiding the search for native-like protein conformations with an ab-initio tree-based exploration. Int J Robot Res. 2010;29(8):1106–27.

Siew N, Elofsson A, Rychlewski L, Fischer D. MaxSub: an automated measure for the assessment of protein structure prediction quality. Bioinformatics. 2000;16(9):776–85.

Uziela K, Shu N, Wallner B, Elofsson A. ProQ3: improved model quality assessments using Rosetta energy terms. Sci Rep. 2016;6(1):33509.

Varadi M, Anyango S, Deshpande M, Nair S, Natassia C, Yordanova G, et al. AlphaFold protein structure database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. Nucleic Acids Res. 2021;50(D1):D439–44. https://doi.org/10.1093/nar/gkab1061

Wallner B, Elofsson A. Can correct protein models be identified? Protein Sci. 2003;12(5):1073–86.

Yang J, Anishchenko I, Park H, Peng Z, Ovchinnikov S, Baker D. Improved protein structure prediction using predicted interresidue orientations. Proc Natl Acad Sci. 2020;117(3):1496–503.

Zaidi S, Schaarschmidt M, Martens J, Kim H, Teh YW, Sanchez-Gonzalez A, et al. Pre-training via denoising for molecular property prediction. The eleventh international conference on learning representations, The International Conference on Learning Representations, Appleton, WI; 2023. p. 26. https://openreview.net/forum?id=tYIMtogyee

How to cite this article: Morehead A, Liu J, Cheng J. Protein structure accuracy estimation using geometry-complete perceptron networks. Protein Science. 2024;33(3):e4932. https://doi.org/10.1002/pro.4932