

RESEARCH

Open Access



End-to-end protein–ligand complex structure generation with diffusion-based generative models

Shuya Nakata^{1*} , Yoshiharu Mori^{1*} and Shigenori Tanaka^{1*}

*Correspondence:
shuyanakata@gmail.com;
ymori@landscape.kobe-u.ac.jp;
tanaka2@kobe-u.ac.jp

¹ Graduate School of System
Informatics, Kobe University,
Kobe, Japan

Abstract

Background: Three-dimensional structures of protein–ligand complexes provide valuable insights into their interactions and are crucial for molecular biological studies and drug design. However, their high-dimensional and multimodal nature hinders end-to-end modeling, and earlier approaches depend inherently on existing protein structures. To overcome these limitations and expand the range of complexes that can be accurately modeled, it is necessary to develop efficient end-to-end methods.

Results: We introduce an equivariant diffusion-based generative model that learns the joint distribution of ligand and protein conformations conditioned on the molecular graph of a ligand and the sequence representation of a protein extracted from a pre-trained protein language model. Benchmark results show that this protein structure-free model is capable of generating diverse structures of protein–ligand complexes, including those with correct binding poses. Further analyses indicate that the proposed end-to-end approach is particularly effective when the ligand-bound protein structure is not available.

Conclusion: The present results demonstrate the effectiveness and generative capability of our end-to-end complex structure modeling framework with diffusion-based generative models. We suppose that this framework will lead to better modeling of protein–ligand complexes, and we expect further improvements and wide applications.

Keywords: Protein–ligand complex, Deep generative model, Molecular interaction, Protein structure prediction

Background

Molecular interactions between proteins and small molecule ligands are fundamental to biological processes, and three-dimensional structures of molecular complexes provide direct insights into their interactions associated with functions [1]. Since experimental structure determination is costly and often challenging, many computational methods have been developed for cheaper and faster modeling.



© The Author(s) 2023, corrected publication 2023. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>. The Creative Commons Public Domain Dedication waiver (<http://creativecommons.org/publicdomain/zero/1.0/>) applies to the data made available in this article, unless otherwise stated in a credit line to the data.

Previous approaches predominantly employ the molecular docking methodology [2–8] that predicts preferred conformations of a ligand in a protein binding site. Despite its success in drug discovery and other applications, correct sampling of binding poses can be limited by poor modeling of protein flexibility [9–11]. Although numerous techniques [12–14] have been proposed to address this problem, they require manual setting with special attention or computationally expensive simulations for conformational sampling. Therefore, modeling the structure of protein–ligand complexes remains a major challenge, especially when the protein structure is flexible or unknown.

Advances in bioinformatics and deep learning have enabled accurate protein structure prediction using multiple sequence alignments (MSAs) and structural templates [15–17]. Subsequent studies have proposed single-sequence structure prediction methods utilizing protein language model (PLM) representations [18–20]. These methods have provided highly accurate structure predictions for proteins that have not been structurally characterized before, opening up new research possibilities. Nevertheless, an end-to-end structure prediction method for protein–ligand complexes that explicitly accounts for ligand features has not yet been established. While using protein structure predictions with docking is a promising approach, current prediction methods lack a principled way to provide diverse models, and obtaining a relevant model suitable for ligand docking is not always possible.

Deep generative modeling [21, 22] is a powerful approach to model high-dimensional and multimodal data distributions and generate samples efficiently. In particular, diffusion-based generative models [23–26] have demonstrated a capacity for high-quality synthesis in many domains, including conformation generation of small molecules [27–30] and proteins [31–33]. A generative model that learns the joint distribution of protein and ligand conformations would enable principled sampling of diverse conformations and provide insights into their ensemble properties.

Recently, several methods have been proposed to apply diffusion-based generative models to protein–ligand complexes [34–36]. For instance, DiffDock [35] modeled the conformation of a ligand relative to a given protein with a diffusion-based generative model. The authors reported significant performance gains over existing methods on the PDBbind [37, 38] benchmark dataset and highlighted the critical issues with regression-based frameworks [39, 40]. In particular, NeuralPLexer [34], similar work to ours, proposed to model the structure of protein–ligand complexes with a hierarchical diffusion model. However, this method still depended on protein backbone templates, and it is unclear whether diffusion models are applicable without structural inputs.

In this work, we propose an end-to-end framework to generate ensembles of protein–ligand complex structures by modeling their probability distributions with equivariant diffusion-based generative models (Fig. 1). By incorporating the essence of the state-of-the-art protein structure prediction methods, the proposed framework can generate diverse structures of protein–ligand complexes without depending on existing protein structures, thus providing a novel and efficient method.

Results

Generative modeling of protein–ligand complex structures

We model the joint distribution of 3D coordinates of protein C_α and ligand non-hydrogen atoms with a variant of equivariant diffusion-based generative model [29]. The

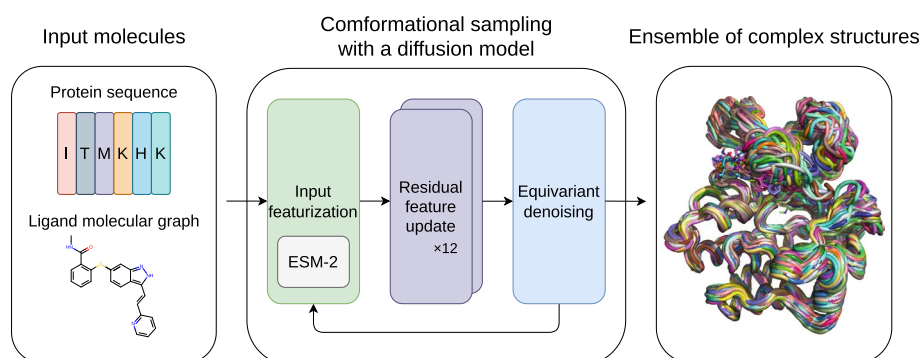


Fig. 1 Overview of the proposed framework. For inputs, protein amino acid sequence and ligand molecular graph are employed. The conformational sampling process involves the iterative application of input featurization, residual feature update, and equivariant denoising to generate an ensemble of complex structures. ESM-2 [18], a large-scale protein language model (PLM), is utilized to featurize input protein sequences

model uses a protein amino acid sequence and a ligand molecular graph as input and iteratively refines the 3D coordinates of the molecules starting from random noises to generate statistically independent structures. We utilized the ESM-2 [18] model, a large-scale PLM used for protein structure prediction, to extract structural and phylogenetic information from the input amino acid sequence.

We used protein–ligand complex structures from the PDBbind [37] database, a collection of biomolecular complexes deposited in the Protein Data Bank (PDB) [41], for training. We employed the time-based split proposed in EquiBind [39], where structures released in 2019 or later were used for evaluation, and any ligand overlap with the test set was removed for training and validation. For computational cost reasons, we only used structures with the number of modeled atoms (the sum of the numbers of protein residues and ligand non-hydrogen atoms) less than or equal to 384. This resulted in 9430 structures for training, 552 for validation, and 207 for evaluation. We used the Adam [42] optimizer with the base learning rate 4×10^{-4} , $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$ and linearly increased the learning rate over the first 1000 optimization steps. We trained our model for around 150 epochs using a mini-batch size of 24. For evaluation, we used an exponential moving average of our parameters with the best validation loss, calculated with a decay rate of 0.999.

Though we designed our model to be protein structure-free, we also trained another version of the model that accepts protein backbone templates as input for reference. We denote the original protein structure-free model as DPL (Diffusion model for Protein–Ligand complexes) and this protein structure-dependent version as DPL+S.

Benchmark on the PDBbind test set

To assess the generative capability of our models in terms of the reproducibility of the experimentally observed protein conformations and ligand binding poses, we sampled 64 structures for each complex in the PDBbind test set and compared them with the PDB-registered structures. We used TM-align [43] for structural alignment with experimental structures and used the resulting TM-score to evaluate protein conformations. For the evaluation of ligand binding poses, we calculated the heavy-atom

root mean square deviation between generated and experimental ligands (L-rms) after aligning the proteins. As a baseline, molecular docking methods GNINA [7] and AutoDock Vina 1.2.0 [8] were used in blind self-docking settings with default parameters, except we increased exhaustiveness from 8 to 64 and num_modes to 64.

Figure 2A shows the median modeling accuracies for each complex in the PDBbind test set. From this data, we can see that our models successfully reproduced the experimentally observed protein conformations and ligand binding poses for a significant number of complexes. Even though DPL did not use protein structures as inputs, it was able to sample a complex structure in which the protein conformation was close to the experimental one (TM-score > 0.9) for more than 70% of the complexes

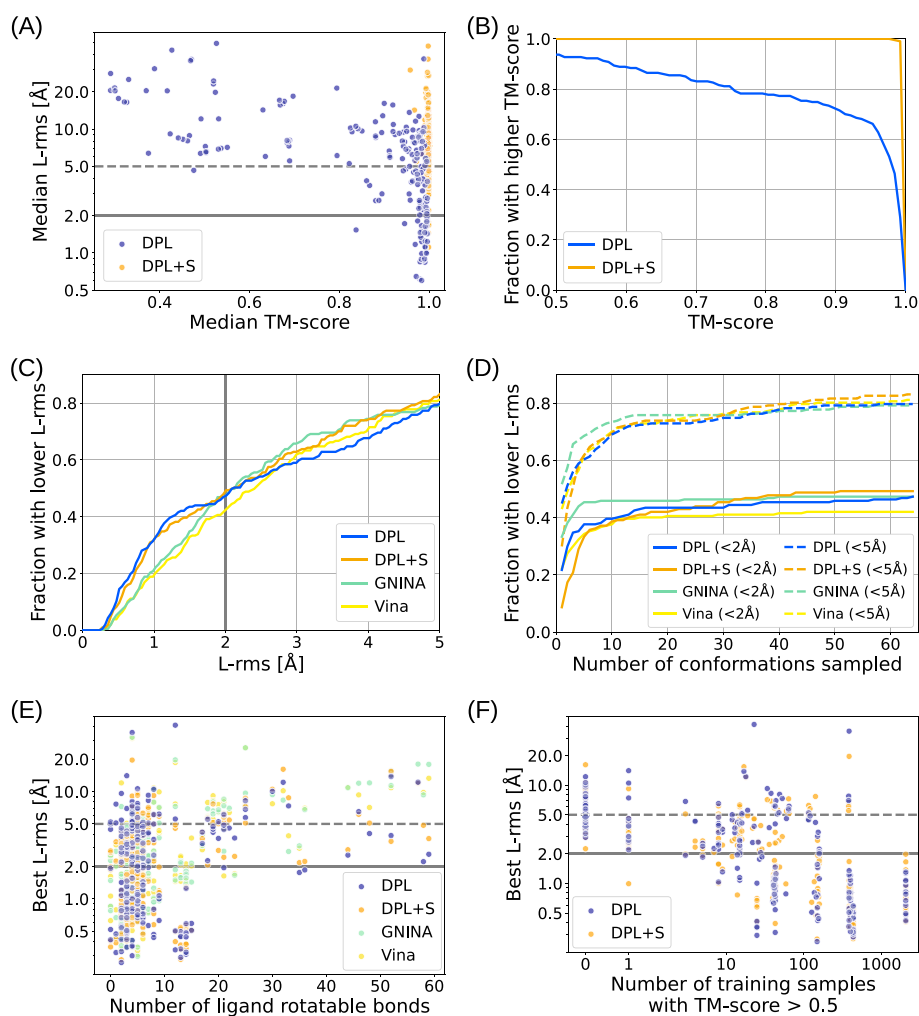


Fig. 2 Benchmark results on the PDBbind test set. **A** Median modeling accuracies for each complex in the test set compared between our models DPL (blue) and DPL+S (orange). **B** Fraction of complexes where a structure with TM-score above the threshold, varying from 0.5 to 1.0, was sampled. **C** Fraction of complexes where a structure with L-rms below the threshold, varying from 0 Å to 5 Å, was sampled. The results obtained with the molecular docking methods (GNINA and AutoDock Vina) are also shown. **D** Performance as a function of the number of generative samples. Two thresholds 2 Å and 5 Å are employed for L-rms. **E** Relation between performance and the number of ligand rotatable bonds. **F** Relation between performance and the number of related training data

in the test set (Fig. 2B). Concerning the binding pose accuracies, our models achieved accuracy comparable to or better than baseline methods, especially in the range of practical importance, where the threshold is less than 2 Å (Fig. 2C). Interestingly, the difference in binding pose accuracy between the two models DPL and DPL+S was insignificant. This is likely attributed to the fact that, as inferred from Fig. 2A, generating the protein structure is almost always feasible if the model understands the complex well enough to reproduce the correct binding pose. Figure 2D shows how the performances change with the number of generative samples. Because our models generated diverse structures, a reasonable number of samples was needed to reproduce the experimental binding poses, while the performance approximately saturated after a few dozen samples. The relationships between the binding pose accuracy and the number of ligand rotatable bonds and that of related structures in the training set are shown in Figs. 2E and F, respectively. Although the binding pose accuracy decreased as the ligand size increased, our models performed better for larger ligands than the baseline methods (Fig. 2E), indicating their ability to efficiently handle many degrees of freedom. Besides, our models were able to sample more accurate binding poses for complexes with more related training samples (Fig. 2F). This indicates that the modeling accuracy is dependent on the training data, suggesting that the performance could be improved by enriching the training data or removing any existing biases in order to prevent overfitting (as discussed in the following section).

Effectiveness of protein structure-free modeling

In this experiment, we validated the effectiveness of protein structure-free modeling in a more challenging situation where ligand-bound protein structure was not available. We performed conformational sampling on 23 complexes from the PocketMiner [44] dataset, a collection of apo-holo protein structure pairs with significant conformational changes upon ligand binding. These 23 complexes were selected by eliminating those with multiple annotated ligands annotated and those used for the training from the original dataset of 38 complexes. For methods that require protein structures as input, we examined both cases using the apo and holo structures.

Figure 3 shows the modeling accuracies on the PocketMiner dataset. As can be seen from Fig. 3A and B, the performance of our models was not so good as for the PDBbind test set, and was worse than the docking with the holo structures. A possible reason for this might be that nearly half of the complexes (11 of the 23) in the PocketMiner dataset were out-of-distribution, i.e., no related samples (TM-score > 0.5) were observed during the training. Nevertheless, DPL outperformed both DPL+S and GNINA when the holo structures were unavailable (Fig. 3B), as the performance of these structure-dependent methods was significantly degraded using apo structures (Fig. 3C). In fact, DPL was able to sample structures with L-rms less than 5 Å on 10 of the 17 complexes where the docking with the apo structure failed. It is also worth noting that the superior performance of DPL was observed even on the out-of-distribution complexes (Fig. 3C).

An example of a Casein kinase II subunit alpha with an inhibitor bound to the substrate binding site (PDB ID 5OSZ) is presented in Fig. 4. In this example, the correct binding pose could not be sampled by the docking with the apo structure because its binding site conformation was unsuitable for ligand binding (Fig. 4C). In contrast,

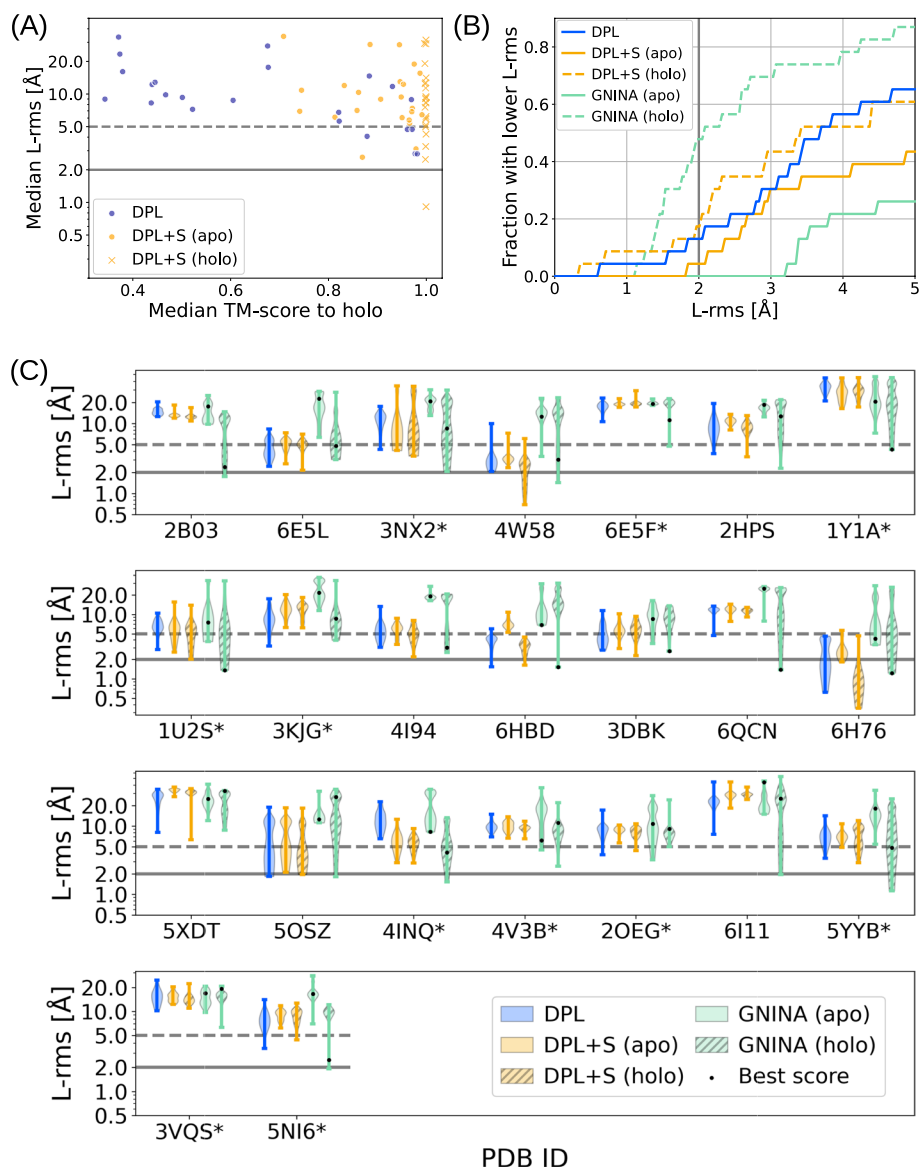


Fig. 3 Modeling accuracies on the PocketMiner dataset. **A** Median modeling accuracies for each complex in the PocketMiner dataset, where the results by DPL, DPL+S (apo), and DPL+S (holo) are compared. **B** Fraction of complexes where a structure with L-rms below the threshold, varying from 0 Å to 5 Å, was sampled. The results obtained with GNINA are also shown for the apo and holo structures. **C** Distribution of L-rms for each complex in the PocketMiner dataset. The * on the PDB ID indicates that no related sample (TM-score > 0.5) was observed during the training

DPL successfully generated proper complex structures without using the holo structure (Fig. 4A and D), demonstrating its effectiveness.

Figure 5 illustrates the distribution of protein conformations generated by DPL for the complex of Sucrose-phosphatase and α -D-glucose (PDB ID: 1U2S). Even though no structure related to this complex was observed during the training, DPL was able to sample diverse protein conformations (Fig. 5D), including both holo-like (Fig. 5A) and apo-like (Fig. 5C) ones. From the data in Fig. 5D, we can also see that the input

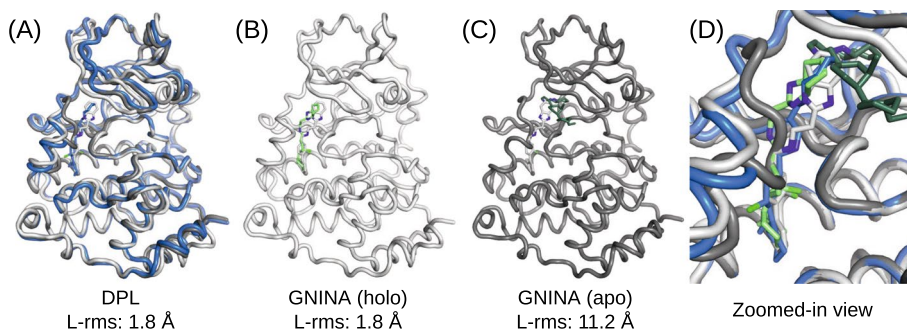


Fig. 4 An example of a Casein kinase II subunit alpha with an inhibitor bound around the substrate binding site. The PDB-registered holo structure is shown in light gray (PDB ID: 5OSZ), and the apo structure is shown in dark gray (PDB ID: 6YPK). **A** Best L-rms complex structure generated by DPL (blue). **B** Best L-rms binding pose generated by GNINA using the holo structure (light green). **C** Best L-rms binding pose generated by GNINA using the apo structure (dark green). **D** Superimposition of all structures zoomed in on the ligand binding site

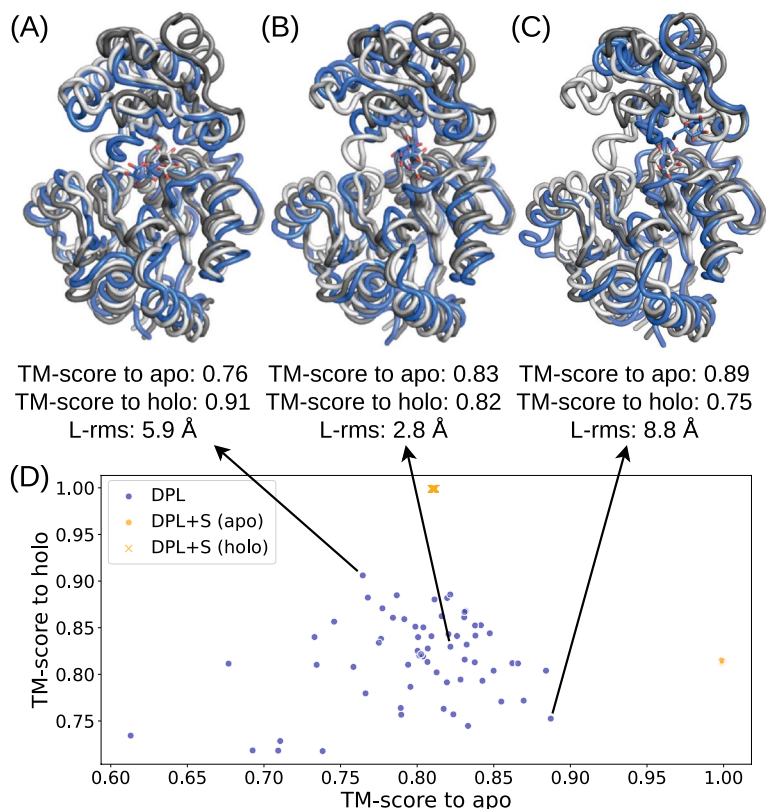


Fig. 5 Distributions of protein conformations generated by DPL for the complex of Sucrose-phosphatase and α -D-glucose. The PDB-registered holo structure is shown in light gray (PDB ID: 1U2S), the apo structure is shown in dark gray (PDB ID: 1S2O), and the generated structure is shown in blue. **A** Generated structure closest to the holo structure. **B** Best L-rms generated structure. **C** Generated structure closest to the apo structure. **D** Distributions of the generated protein conformations projected onto the two-dimensional plane of TM-score to apo and TM-score to holo (blue symbols). The conformations obtained with DPL+S are also shown (orange symbols)

of protein structures biased the generated conformations so strongly that only small fluctuations were observed.

Discussion

The present results highlight the critical problem with previous protein structure-dependent approaches. Namely, preparing a protein structure suitable for ligand binding prior to determining the complex structure is an ill-posed problem, even though the sampling of binding poses is sensitive to inputs in these methods. Our approach aims to overcome this problem by generating the structure of protein–ligand complexes end-to-end without being biased by the input of protein structures.

Our protein structure-free model, DPL, demonstrated the ability to sample complex structures in many protein–ligand complexes with binding poses comparable to those obtained by molecular docking. Furthermore, it is encouraging that our model was able to generalize to some of the proteins for which no related structure was used for training, and successfully sampled binding poses that were inaccessible by docking methods. However, its performance was still limited for complexes with small amounts of related training data, raising concerns about overfitting to the protein structures used in the training data (see also Figure S2 in Additional file 1). Because the time-based split [39] of the PDBbind dataset used in this study does not account for the overlap and redundancy in protein sequences and structures, the performance of our model could be diminished if we used a dataset constructed on the basis of these criteria. Testing the generalizability for new proteins more rigorously using datasets based on those established criteria in protein biology as in CASP [45] is an important direction for future research.

One approach to addressing overfitting and improving performance on new complexes is to enhance the dataset and reduce any existing biases. The dataset used in this study was limited in size due to the application of a cutoff based on the total number of atoms. It would be beneficial to randomly crop partial structures from large complexes for training purposes, as is performed in protein structure prediction studies [17, 46]. We provide an example of such a procedure in Additional file 1 (Algorithm S1), although it was not used in the experiments performed in the present study. Furthermore, in our training dataset, some complexes had a thousand times more structures than others, which could hinder generalization. This bias could be eliminated by clustering complex structures based on protein sequence or structural similarity. Recent studies in protein structure prediction have reported improvements in prediction accuracy and generalizability through refinement of model architecture [17], training data [47], and scaling of pre-trained protein language models [18]. In light of these reports and our observations, we are optimistic that modeling accuracy and generalizability will improve with better training data and the development of larger or more sophisticated models.

Although we do not consider that our model outperforms the state-of-the-art protein structure prediction methods in terms of the protein structure prediction accuracy, the conformational diversity observed in our experiments is remarkable given the current challenge of multi-state sampling in these methods. These methods rely on *ad hoc* techniques such as MSA subsampling, changing the number of recycles, and enabling drop-out during inference to sample multiple structures [47]. In contrast, our model was able to sample diverse protein structures in a principled manner, without relying on these

techniques. We suppose this is due to our generative approach, which models the data distribution, as opposed to a regression approach, which predicts a single output for each input. The effectiveness of the generative approach has also been discussed in the context of ligand docking [35].

In this study, we focused on the generation of complex structures and left the scoring of their binding poses, which is essential for practical applications such as drug discovery, as a task for future work. To address this limitation, we discuss three approaches for scoring the binding poses below. One possible scoring metric could involve building an all-atom model of the generated structure and evaluating it using established metrics like docking scores. Taking advantage of physics-based methods such as Rosetta [48], this approach could help filter out inaccurate structures, and provide a small number of the most promising structures. However, this approach can complicate the procedure and increase computational costs. Another approach may be to train a regression model that estimates the confidence measure of the generated structure. For example, DiffDock [35] trains a confidence model that predicts whether each binding pose has an L-rms below 2 Å and uses this model to rank the generated poses. The training data for the confidence model can be generated by using a learned diffusion model. Although this approach requires training of the confidence model, which could potentially introduce additional bias, it would be a practical solution for efficiently ranking large numbers of candidate structures. Furthermore, as a third approach, an interesting avenue for future research would be to develop a scoring method that makes use of the learned diffusion model. The diffusion model used in this study is a likelihood-based generative model that is capable of estimating the likelihood of each sample [15, 29]. Since the likelihood is affected by the overall structure, it would be necessary to devise a new method for scoring the binding poses. To correctly rank the generated structures, it may also be important to improve the model or likelihood estimation methods to reduce the variance of the estimate.

An attractive approach to improving the method presented in this study is to explicitly model all the heavy atoms of the protein, as this is essential for interpreting the generated structures and combining them with other tools. Our model may be extended to the atomic level with little increase in computational cost by incorporating backbone orientations and side-chain dihedral angles into the diffusion model [32]. An alternative approach would be to utilize existing methods such as Rosetta [48] to construct an all-atom model of the protein from the structure generated by our model. Our results on the physical accuracy of the generated structures provided in Additional file 1 (Figure S1) suggest that the structures generated by our model are physically plausible, and a reasonable result could be obtained by building an all-atom model and optimizing its structure.

Conclusion

We have introduced an equivariant diffusion-based generative model for end-to-end protein–ligand complex structure generation to overcome the limitations of previous structure-dependent approaches. The structures generated by our model were diverse and included those with proper protein conformations and ligand binding poses. When the ligand-bound protein structures were not available, our protein structure-free model

showed better binding pose accuracy than the protein structure-dependent model and docking method, demonstrating the effectiveness of our end-to-end approach. While the performance was limited for complexes with small amounts of related training data, it is encouraging that generalization was still observed in some of the new complexes. Based on these promising results, we conclude that the proposed framework will lead to better modeling of protein–ligand complexes, and we expect further improvements and wide applications.

Methods

In this section, we describe the formulation of the equivariant diffusion-based model used in our framework, which is based on Variational Diffusion Models [26] and E(3) Equivariant Diffusion Models [29].

The diffusion process

We first define an equivariant diffusion process for atom coordinates \mathbf{x} and a sequence of increasingly noisy versions of \mathbf{x} called latent variables \mathbf{z}_t , where t ranges from $t = 0$ to $t = 1$ (Fig. 6). To ensure that the distributions are invariant to translation, we use distributions on the linear subspace [28] where the centroid of the molecular system is always at the origin and define \mathcal{N}_x as a Gaussian distribution on this subspace, following [29]. The distribution of latent variable \mathbf{z}_t conditioned on \mathbf{x} , for any $t \in [0, 1]$ is given by

$$q(\mathbf{z}_t|\mathbf{x}) = \mathcal{N}_x(\alpha_t\mathbf{x}, \sigma_t^2\mathbf{I}), \tag{1}$$

where α_t and σ_t^2 are strictly positive scalar-valued functions of t that control how much signal is retained and how much noise is added, respectively. We use the variance preserving process [23, 24] where $\alpha_t = \sqrt{1 - \sigma_t^2}$, and assume that α_t is a smooth and monotonically decreasing function of t satisfying $\alpha_0 \approx 1$ and $\alpha_1 \approx 0$. Since this diffusion process is Markovian, it can also be written using transition distributions as

$$q(\mathbf{z}_t|\mathbf{z}_s) = \mathcal{N}_x(\alpha_{t|s}\mathbf{z}_s, \sigma_{t|s}^2\mathbf{I}) \tag{2}$$

for any $t > s$ with $\alpha_{t|s} = \alpha_t/\alpha_s$ and $\sigma_{t|s}^2 = \sigma_t^2 - \alpha_{t|s}^2\sigma_s^2$. The posterior of the transitions given \mathbf{x} is also Gaussian and can be obtained using Bayes' rule:

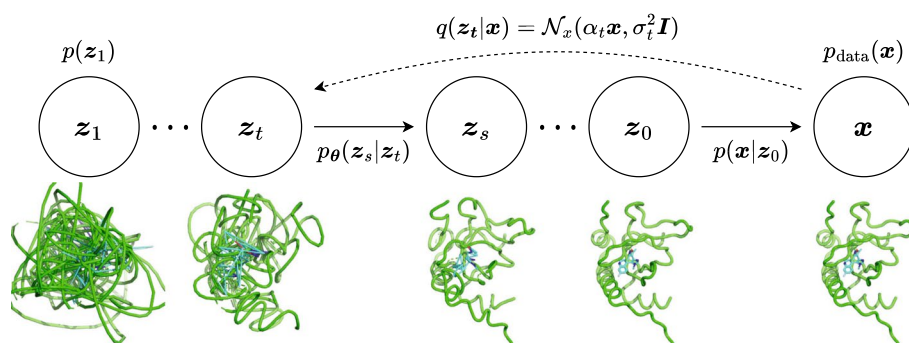


Fig. 6 The diffusion and generative denoising process

$$q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) = \mathcal{N}_x(\boldsymbol{\mu}_{t \rightarrow s}(\mathbf{z}_t, \mathbf{x}), \sigma_{t \rightarrow s}^2 \mathbf{I}), \tag{3}$$

where

$$\boldsymbol{\mu}_{t \rightarrow s} = \frac{\alpha_{t|s}\sigma_s^2}{\sigma_t^2} \mathbf{z}_t + \frac{\alpha_s\sigma_t^2}{\sigma_t^2} \mathbf{x}, \quad \sigma_{t \rightarrow s} = \frac{\sigma_{t|s}\sigma_s}{\sigma_t}. \tag{4}$$

The generative denoising process

The generative model is defined by the inverse of the diffusion process, where a sequence of latent variables \mathbf{z}_t is sampled backward in time from $t = 1$ to $t = 0$. Discretizing time uniformly into T timesteps, we can define the generative model as

$$p_\theta(\mathbf{x}) = \int_{\mathbf{z}} p(\mathbf{z}_1)p(\mathbf{x}|\mathbf{z}_0) \prod_{i=1}^T p_\theta(\mathbf{z}_{s(i)}|\mathbf{z}_{t(i)}), \tag{5}$$

where $s(i) = (i - 1)/T$ and $t(i) = i/T$. The variance preserving specification and the assumption that $\alpha_1 \approx 0$ allow us to assume that $q(\mathbf{z}_1) = \mathcal{N}_x(\mathbf{0}, \mathbf{I})$. We thus model the marginal distribution of \mathbf{z}_1 as a standard Gaussian:

$$p(\mathbf{z}_1) = \mathcal{N}_x(\mathbf{0}, \mathbf{I}). \tag{6}$$

Similarly, with the variance preserving specification and the assumption that $\alpha_0 \approx 1$, we can assume that $q(\mathbf{z}_0|\mathbf{x})$ is a highly peaked distribution and $p_{\text{data}}(\mathbf{x})$ can be approximated as constant over this narrow peak. Therefore we have

$$q(\mathbf{x}|\mathbf{z}_0) = \frac{q(\mathbf{z}_0|\mathbf{x})p_{\text{data}}(\mathbf{x})}{\int_{\tilde{\mathbf{x}}} q(\mathbf{z}_0|\tilde{\mathbf{x}})p_{\text{data}}(\tilde{\mathbf{x}})} \approx \frac{q(\mathbf{z}_0|\mathbf{x})}{\int_{\tilde{\mathbf{x}}} q(\mathbf{z}_0|\tilde{\mathbf{x}})} = \mathcal{N}_x(\mathbf{x}|\mathbf{z}_0/\alpha_0, \sigma_0^2/\alpha_0^2 \mathbf{I}). \tag{7}$$

We then model $q(\mathbf{x}|\mathbf{z}_0)$ as

$$p(\mathbf{x}|\mathbf{z}_0) = \mathcal{N}_x(\mathbf{x}|\mathbf{z}_0/\alpha_0, \sigma_0^2/\alpha_0^2 \mathbf{I}). \tag{8}$$

Finally, we define the conditional model distributions as

$$p_\theta(\mathbf{z}_s|\mathbf{z}_t) = q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x} = \hat{\mathbf{x}}_\theta(\mathbf{z}_t; t)), \tag{9}$$

which is equivalent to $q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x})$, but with the original coordinates \mathbf{x} being replaced by the output of a time-dependent denoising model $\hat{\mathbf{x}}_\theta(\mathbf{z}_t; t)$ that predicts \mathbf{x} from its noisy version \mathbf{z}_t using a neural network with parameter θ . In practice, the denoising model is parametrized in terms of a noise prediction model $\hat{\boldsymbol{\epsilon}}_\theta(\mathbf{z}_t; t)$:

$$\hat{\mathbf{x}}_\theta(\mathbf{z}_t; t) = (\mathbf{z}_t - \sigma_t \hat{\boldsymbol{\epsilon}}_\theta(\mathbf{z}_t; t))/\alpha_t. \tag{10}$$

With this parameterization, $\boldsymbol{\mu}_{t \rightarrow s}(\mathbf{z}_t, \hat{\mathbf{x}}_\theta(\mathbf{z}_t; t))$ can be calculated as

$$\begin{aligned} \mu_{t \rightarrow s}(\mathbf{z}_t, \hat{\mathbf{x}}_{\theta}(\mathbf{z}_t; t)) &= \frac{\alpha_{t|s}\sigma_s^2}{\sigma_t^2} \mathbf{z}_t + \frac{\alpha_s\sigma_{t|s}^2}{\sigma_t^2} \mathbf{x} \\ &= \frac{1}{\alpha_{t|s}} \mathbf{z}_t - \frac{\sigma_{t|s}^2}{\alpha_{t|s}\sigma_t} \hat{\boldsymbol{\epsilon}}_{\theta}(\mathbf{z}_t; t). \end{aligned} \tag{11}$$

We can generate samples via ancestral sampling from this distribution (Algorithm 1).

Algorithm 1 Sample generation with noise prediction model $\hat{\boldsymbol{\epsilon}}_{\theta}$.

```

procedure GENERATE(Number of timesteps  $T = 64$ )
  Sample  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 
  Subtract centroid from  $\mathbf{z}$ 
  for  $i$  in  $T, T - 1, \dots, 1$  do
    Sample  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 
    Subtract centroid from  $\boldsymbol{\epsilon}$ 
    Compute  $s = (i - 1)/T, t = i/T$ 
    Compute  $\mathbf{z} = \frac{1}{\alpha_{t|s}} \mathbf{z} - \frac{\sigma_{t|s}^2}{\alpha_{t|s}\sigma_t} \hat{\boldsymbol{\epsilon}}_{\theta}(\mathbf{z}; t) + \sigma_{t \rightarrow s} \cdot \boldsymbol{\epsilon}$ 
  end for
  Sample  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 
  Subtract centroid from  $\boldsymbol{\epsilon}$ 
   $\mathbf{x} = (\mathbf{z} + \sigma_0 \boldsymbol{\epsilon})/\alpha_0$ 
  return  $\mathbf{x}$ 
end procedure

```

Optimization objective

We optimize the parameters θ toward the variational lower bound (VLB) of the marginal likelihood, which is given by

$$\begin{aligned} -\log p_{\theta}(\mathbf{x}) &\leq -\text{VLB}_{\theta}(\mathbf{x}) \\ &= \underbrace{D_{KL}[q(\mathbf{z}_T|\mathbf{x})\|p(\mathbf{z}_T)]}_{\text{Prior loss}} + \underbrace{\mathbb{E}_{\mathbf{z}_0 \sim q(\mathbf{z}_0|\mathbf{x})}[-\log p(\mathbf{x}|\mathbf{z}_0)]}_{\text{Reconstruction loss}} + \underbrace{\mathcal{L}_T(\mathbf{x})}_{\text{Diffusion loss}}, \end{aligned} \tag{12}$$

where $\mathcal{L}_T(\mathbf{x}) = \sum_{i=1}^T \mathbb{E}_{\mathbf{z}_{t(i)} \sim q(\mathbf{z}_{t(i)}|\mathbf{x})} [D_{KL}[q(\mathbf{z}_{s(i)}|\mathbf{z}_{t(i)}, \mathbf{x})\|p_{\theta}(\mathbf{z}_{s(i)}|\mathbf{z}_{t(i)})]]$ and D_{KL} refers to the Kullback-Leibler divergence. Since $p(\mathbf{z}_T)$ and $p(\mathbf{z}_0|\mathbf{x})$ contain no learnable parameter in our parameterization, the model is optimized by minimizing the third term, diffusion loss. As shown in [26], if we define the signal-to-noise ratio (SNR) at time t as $\text{SNR}(t) = \alpha_t^2/\sigma_t^2$ and $\gamma(t) = -\log \text{SNR}(t)$, then the diffusion loss can be simplified to

$$\begin{aligned} \mathcal{L}_T(\mathbf{x}) &= \sum_{i=1}^T \mathbb{E}_{\mathbf{z}_t \sim q(\mathbf{z}_t|\mathbf{x})} [D_{KL}[q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x})\|p_{\theta}(\mathbf{z}_s|\mathbf{z}_t)]] \\ &= \frac{1}{2} \sum_{i=1}^T \mathbb{E}_{\mathbf{z}_t \sim q(\mathbf{z}_t|\mathbf{x})} \left[(\text{SNR}(s) - \text{SNR}(t)) \|\mathbf{x} - \hat{\mathbf{x}}_{\theta}(\mathbf{z}_t; t)\|^2 \right] \\ &= \frac{1}{2} \sum_{i=1}^T \mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}_x(\mathbf{0}, \mathbf{I})} \left[(\exp(\gamma(s) - \gamma(t)) - 1) \|\boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}}_{\theta}(\mathbf{z}_t; t)\|^2 \right], \end{aligned} \tag{13}$$

where $s = (i - 1)/T$, $t = i/T$, and $\mathbf{z}_t = \alpha_t \mathbf{x} + \sigma_t \epsilon$. Furthermore, we can consider a continuous time model corresponding to $T \rightarrow \infty$. In the limit of $T \rightarrow \infty$, the diffusion loss becomes

$$\begin{aligned} \mathcal{L}_\infty(\mathbf{x}) &= \frac{1}{2} \mathbb{E}_{\epsilon \sim \mathcal{N}_x(\mathbf{0}, \mathbf{I})} \int_0^1 \gamma'(t) \|\epsilon - \hat{\epsilon}_\theta(\mathbf{z}_t; t)\|^2 dt \\ &= \frac{1}{2} \mathbb{E}_{\epsilon \sim \mathcal{N}_x(\mathbf{0}, \mathbf{I}), t \sim \mathcal{U}(0,1)} \left[\gamma'(t) \|\epsilon - \hat{\epsilon}_\theta(\mathbf{z}_t; t)\|^2 \right], \end{aligned} \tag{14}$$

where $\gamma'(t) = d\gamma(t)/dt$. We use the Monte Carlo estimator of this continuous loss for parameter optimization (Algorithm 2).

Algorithm 2 Parameter optimization of noise prediction model $\hat{\epsilon}_\theta$.

```

procedure OPTIMIZE(Data point  $x$ )
  Sample  $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ ,  $t \sim \mathcal{U}(1, T)$ 
  Subtract center of gravity from  $\epsilon$ 
  Compute  $\mathbf{z} = \alpha_t \mathbf{x} + \sigma_t \epsilon$ 
  Minimize  $\frac{1}{2} \gamma'(t) \|\epsilon - \hat{\epsilon}_\theta(\mathbf{z}; t)\|^2$ 
end procedure
    
```

Model architecture

As illustrated in Fig. 7, our noise prediction model consists of three procedures: (1) input featurization, (2) residual feature update, and (3) equivariant denoising. In this section, we outline each of these procedures, which are also described in Algorithm 3. For extensive details on the experimental setup, data, hyperparameters, and implementation, please refer to our code available at <https://github.com/shuyana/DiffusionProteinLigand>.

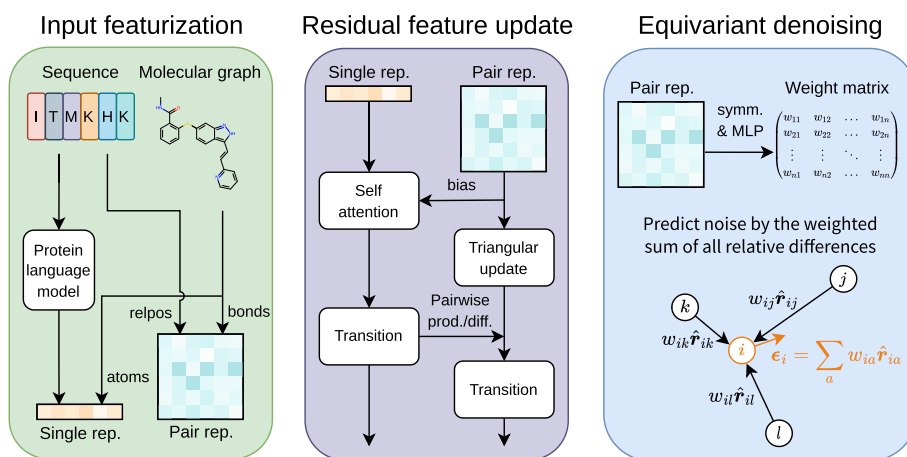


Fig. 7 Overview of the model architecture. In the process of input featurization, single and pair representations are constructed. These features are then iteratively updated by the Folding blocks. The final pair representation is transformed by an MLP into a weight matrix to predict the denoising vector

Input featurization

We construct a single representation and a pair representation from a protein amino acid sequence and a ligand molecular graph. We utilize the 650 M parameters ESM-2 [18] model, a large-scale protein language model pre-trained on ~ 65 million unique protein sequences from the UniRef [49] database, to extract structural and phylogenetic information from amino acid sequences. To create the single representation of proteins, the final layer of the ESM-2 model is linearly mapped after normalization and then added to the amino acid embeddings. For the pair representation of proteins, we use the pairwise relative positional encoding described in the literature [17]. The representations of ligands are constructed through feature embedding of atoms and bonds. The features of the ligand atoms include: atomic number; chirality; degree; formal charge; the number of connected hydrogens; the number of radical electrons; hybridization type; whether or not it is aromatic; and whether or not it is in a ring. For ligand bonds, we use three features: bond type; stereo configuration; and whether or not it is considered to be conjugated. We concatenate the protein and ligand representations and then add them to the radial basis embeddings of atom distances and the sinusoidal embedding of diffusion time to obtain the initial representations of complexes.

Residual feature update

We jointly update the single and pair representations with the 12 Folding blocks described in the ESMFold [18]. The Folding Block is a single-sequence version of the Evoformer used in AlphaFold2 [17]. It updates the single and pair representations in a residual manner, wherein the two representations mutually influence one another. The triangular update module within the Folding Block was designed to predict proximity in three-dimensional space, inspired by the need for consistent pair representations, such as the triangle inequality on distances. Although the Folding block was originally developed for proteins, it can also be used for protein–ligand systems without architectural modification.

Equivariant denoising

In the process of equivariant denoising, the final pair representation is symmetrized and transformed by a multi-layer perceptron (MLP) into a weight matrix \mathbf{W} . This matrix is used to compute the weighted sum of all relative differences in 3D space for each atom:

$$\hat{\mathbf{e}}_i(\mathbf{z}) = \sum_j \frac{W_{ij}(\mathbf{z})}{\|\mathbf{z}_i - \mathbf{z}_j\|} \cdot (\mathbf{z}_i - \mathbf{z}_j). \quad (15)$$

The centroid is then removed from this, resulting in the output of our noise prediction model $\hat{\mathbf{e}}$.

Finally, we note that the model described above is SE(3)-equivariant, that is,

$$\begin{aligned}
\hat{\epsilon}_i(\mathbf{Rz} + \mathbf{t}) &= \sum_j \frac{W_{ij}(\mathbf{Rz} + \mathbf{t})}{\|(\mathbf{Rz}_i + \mathbf{t}) - (\mathbf{Rz}_j + \mathbf{t})\|} \cdot ((\mathbf{Rz}_i + \mathbf{t}) - (\mathbf{Rz}_j + \mathbf{t})) \\
&= \mathbf{R} \sum_j \frac{W_{ij}(\mathbf{Rz} + \mathbf{t})}{\|\mathbf{z}_i - \mathbf{z}_j\|} \cdot (\mathbf{z}_i - \mathbf{z}_j) \\
&= \mathbf{R} \sum_j \frac{W_{ij}(\mathbf{z})}{\|\mathbf{z}_i - \mathbf{z}_j\|} \cdot (\mathbf{z}_i - \mathbf{z}_j) \\
&= \mathbf{R}\hat{\epsilon}_i(\mathbf{z})
\end{aligned} \tag{16}$$

for any rotation \mathbf{R} and translation \mathbf{t} . The second last equation holds because the final representation, and thus, the weight matrix \mathbf{W} , depend on atom coordinates only through atom distances that are invariant to rotation and translation.

Algorithm 3 Noise prediction model. EmbedXXX embeds discrete features in representation space. ESM_LastLayer returns the last hidden representations from an ESM language model.

```

procedure PREDICT(Noisy atom coordinates  $\mathbf{z}$ , diffusion time  $t$ , ligand molecular graph  $\mathcal{G}$ , protein
sequence seq, protein backbone template templ (optional), Number of modeled atoms  $N_{\text{atom}}$ ,  $s \in \mathbb{R}^{N_{\text{atom}} \times C_s}$ ,  $p \in \mathbb{R}^{N_{\text{atom}} \times N_{\text{atom}} \times C_p}$ ,  $w \in \mathbb{R}^{N_{\text{atom}} \times N_{\text{atom}}}$ ,  $C_s = 512$ ,  $C_p = 64$ ,  $N_{\text{block}} = 12$ )
   $s_{\text{protein}} = \text{EmbedAminoAcids}(\text{seq}) + \text{Linear}(\text{LayerNorm}(\text{ESM\_LastLayer}(\text{seq})))$ 
   $p_{\text{protein}} = \text{PairwiseRelativePositionalEncoding}(\text{seq})$ 
  if use_template then
     $p_{\text{protein}} = p_{\text{protein}} + \text{Linear}(\text{OneHot}(\text{PseudoC}_{\beta}\text{Distogram}(\text{templ})))$ 
  end if
   $s_{\text{ligand}} = \text{EmbedLigandAtoms}(\mathcal{G})$ 
   $p_{\text{ligand}} = \text{EmbedLigandBonds}(\mathcal{G})$ 
   $s = \text{Concat}([s_{\text{protein}}, s_{\text{ligand}}])$ 
   $p = \text{BlockDiag}([p_{\text{protein}}, p_{\text{ligand}}])$ 
   $p = p + \text{Linear}(\text{RadialBasisDistanceEncoding}(\mathbf{z}))$ 
   $p = p + \text{Linear}(\text{SinusoidalEncoding}(t))$ 
  for  $n$  in  $1, \dots, N$  do
     $s, p = \text{FoldingBlock}_n(s, p)$ 
  end for
   $p = \frac{1}{2}(p + \text{Transpose}(p))$ 
   $w = \text{Linear}(\text{ReLU}(\text{Linear}(\text{LayerNorm}(p))))$ 
   $\hat{\epsilon}_i = \sum_j \frac{w_{ij}}{\|\mathbf{z}_i - \mathbf{z}_j\|} (\mathbf{z}_i - \mathbf{z}_j)$ 
  Subtract centroid from  $\hat{\epsilon}$ 
  return  $\hat{\epsilon}$ 
end procedure

```

Abbreviations

MSA	Multiple sequence alignment
PLM	Protein language model
PDB	Protein data bank
DPL	Diffusion model for protein–ligand complexes
L-rms	Ligand root mean square deviation
SNR	Signal-to-noise ratio
MLP	Multi-layer perceptron

Supplementary Information

The online version contains supplementary material available at <https://doi.org/10.1186/s12859-023-05354-5>.

Additional file 1. Supplementary information.

Acknowledgements

Not applicable.

Author contributions

SN analyzed and interpreted the numerical data in collaboration with YM and ST, and was a major contributor in writing the manuscript. All authors read and approved the final manuscript.

Funding

We acknowledge the Grants-in-Aid for Scientific Research (No. 21K06098) from the Ministry of Education, Culture, Sports, Science and Technology (MEXT), Japan, and MEXT Quantum Leap Flagship Program (No. JPMXS0120330644).

Availability of data and materials

The datasets generated and/or analysed during the current study are available in the GitHub repository, <https://github.com/shuyana/DiffusionProteinLigand>.

Declarations

Ethics approval and consent to participate

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Consent for publication

Not applicable.

Received: 22 December 2022 Accepted: 25 May 2023

Published: 5 June 2023

References

1. Babine RE, Bender SL. Molecular recognition of protein–ligand complexes: applications to drug design. *Chem Rev.* 1997;97(5):1359–472. <https://doi.org/10.1021/cr960370z>.
2. Brooijmans N, Kuntz ID. Molecular recognition and docking algorithms. *Annu Rev Biophys Biom.* 2003;32(1):335–73. <https://doi.org/10.1146/annurev.biophys.32.110601.142532>.
3. Friesner RA, Banks JL, Murphy RB, Halgren TA, Klicic JJ, Mainz DT, Repasky MP, Knoll EH, Shelley M, Perry JK, Shaw DE, Francis P, Shenkin PS. Glide: A new approach for rapid, accurate docking and scoring. 1. method and assessment of docking accuracy. *J Med Chem.* 2004;47(7):1739–49. <https://doi.org/10.1021/jm0306430>.
4. Halgren TA, Murphy RB, Friesner RA, Beard HS, Frye LL, Pollard WT, Banks JL. Glide: A new approach for rapid, accurate docking and scoring. 2. enrichment factors in database screening. *J Med Chem.* 2004;47(7):1750–9. <https://doi.org/10.1021/jm030644s>.
5. Verdonk ML, Cole JC, Hartshorn MJ, Murray CW, Taylor RD. Improved protein–ligand docking using GOLD. *Proteins: Struct Funct Bioinf.* 2003;52(4):609–23. <https://doi.org/10.1002/prot.10465>.
6. Morris GM, Huey R, Lindstrom W, Sanner MF, Belew RK, Goodsell DS, Olson AJ. AutoDock4 and AutoDockTools4: automated docking with selective receptor flexibility. *J Comput Chem.* 2009;30(16):2785–91. <https://doi.org/10.1002/jcc.21256>.
7. McNutt AT, Francoeur P, Aggarwal R, Masuda T, Meli R, Ragoza M, Sunseri J, Koes DR. GNINA 1.0: Molecular docking with deep learning. *J Cheminf.* 2021;13(1). <https://doi.org/10.1186/s13321-021-00522-2>.
8. Eberhardt J, Santos-Martins D, Tillack AF, Forli S. AutoDock Vina 1.2.0: new docking methods, expanded force field, and python bindings. *J Chem Inf Model.* 2021;61(8):3891–8. <https://doi.org/10.1021/acs.jcim.1c00203>.
9. Sousa SF, Fernandes PA, Ramos MJA. Protein–ligand docking: current status and future challenges. *Proteins: Struct Funct Bioinf.* 2006;65(1):15–26. <https://doi.org/10.1002/prot.21082>.
10. Waszkowycz B, Clark DE, Gancia E. Outstanding challenges in protein–ligand docking and structure-based virtual screening. *WIREs Comput Mol Sci.* 2011;1(2):229–59. <https://doi.org/10.1002/wcms.18>.
11. Pagadala NS, Syed K, Tuszynski J. Software for molecular docking: a review. *Biophys Rev.* 2017;9(2):91–102. <https://doi.org/10.1007/s12551-016-0247-1>.
12. Sherman W, Day T, Jacobson MP, Friesner RA, Farid R. Novel procedure for modeling Ligand/Receptor induced fit effects. *J Med Chem.* 2005;49(2):534–53. <https://doi.org/10.1021/jm050540c>.
13. Huang SY, Zou X. Ensemble docking of multiple protein structures: considering protein structural variations in molecular docking. *Proteins: Struct Funct Bioinf.* 2006;66(2):399–421. <https://doi.org/10.1002/prot.21214>.
14. Shin W-H, Seok C. GalaxyDock: protein–ligand docking with flexible protein side-chains. *J Chem Inf Model.* 2012;52(12):3225–32. <https://doi.org/10.1021/ci300342z>.

15. 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. <https://doi.org/10.1073/pnas.1914677117>.
16. Baek M, DiMaio F, Anishchenko I, Dauparas J, Ovchinnikov S, Lee GR, Wang J, Cong Q, Kinch LN, Schaeffer RD, Millán C, Park H, Adams C, Glassman CR, DeGiovanni A, Pereira JH, Rodrigues AV, van Dijk AA, Ebrecht AC, Opperman DJ, Sagmeister T, Buhlheller C, Pavkov-Keller T, Rathinaswamy MK, Dalwadi U, Yip CK, Burke JE, Garcia KC, Grishin NV, Adams PD, Read RJ, Baker D. Accurate prediction of protein structures and interactions using a three-track neural network. *Science.* 2021;373(6557):871–6. <https://doi.org/10.1126/science.abj8754>.
17. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, Tunyasuvunakool K, Bates R, Židek A, Potapenko A, Bridgland A, Meyer C, Kohl SAA, Ballard AJ, Cowie A, Romera-Paredes B, Nikolov S, Jain R, Adler J, Back T, Petersen S, Reiman D, Clancy E, Zielinski M, Steinegger M, Pacholska M, Berghammer T, Bodenstein S, Silver D, Vinyals O, Senior AW, Kavukcuoglu K, Kohli P, Hassabis D. Highly accurate protein structure prediction with AlphaFold. *Nature.* 2021;596(7873):583–9. <https://doi.org/10.1038/s41586-021-03819-2>.
18. Lin Z, Akin H, Rao R, Hie B, Zhu Z, Lu W, Smetanin N, Verkuil R, Kabeli O, Shmueli Y, dos Santos Costa A, Fazel-Zarandi M, Sercu T, Candido S, Rives A. Evolutionary-scale prediction of atomic level protein structure with a language model. *bioRxiv.* 2022. <https://doi.org/10.1101/2022.07.20.500902>.
19. Wu R, Ding F, Wang R, Shen R, Zhang X, Luo S, Su C, Wu Z, Xie Q, Berger B, Ma J, Peng J. High-resolution de novo structure prediction from primary sequence. *bioRxiv.* 2022. <https://doi.org/10.1101/2022.07.21.500999>.
20. Fang X, Wang F, Liu L, He J, Lin D, Xiang Y, Zhang X, Wu H, Li H, Song L. HelixFold-Single: MSA-free protein structure prediction by using protein language model as an alternative. *arXiv:2207.13921.* 2022. <https://doi.org/10.48550/ARXIV.2207.13921>.
21. Goodfellow I, Bengio Y, Courville A. *Deep learning.* Cambridge: MIT Press; 2016.
22. Kingma DP. *Variational inference & deep learning: a new synthesis.* PhD thesis. 2017.
23. Sohl-Dickstein J, Weiss E, Maheswaranathan N, Ganguli S. Deep unsupervised learning using nonequilibrium thermodynamics. In: *International conference on machine learning*, 2015;pp. 2256–2265. <https://doi.org/10.48550/ARXIV.1503.03585>.
24. Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models. In: *Adv Neural Inform Process Syst.* 2020;33:6840–51. <https://doi.org/10.48550/ARXIV.2006.11239>.
25. Song Y, Sohl-Dickstein J, Kingma DP, Kumar A, Ermon S, Poole B. Score-based generative modeling through stochastic differential equations. In: *International conference on learning representations.* 2020. <https://doi.org/10.48550/ARXIV.2011.13456>.
26. Kingma D, Salimans T, Poole B, Ho J. Variational diffusion models. In: *Adv Neural Inform Process Syst.* 2021;34:21696–707. <https://doi.org/10.48550/ARXIV.2107.00630>.
27. Shi C, Luo S, Xu M, Tang J. Learning gradient fields for molecular conformation generation. In: *International conference on machine learning*, 2021;pp. 9558–9568. <https://doi.org/10.48550/ARXIV.2105.03902>.
28. Xu M, Yu L, Song Y, Shi C, Ermon S, Tang J. GeoDiff: A geometric diffusion model for molecular conformation generation. In: *International conference on learning representations.* 2021. <https://doi.org/10.48550/ARXIV.2203.02923>.
29. Hoogeboom E, Satorras VG, Vignac C, Welling M. Equivariant diffusion for molecule generation in 3D. In: *International conference on machine learning*, 2022;pp. 8867–8887. <https://doi.org/10.48550/ARXIV.2203.17003>.
30. Jing B, Corso G, Barzilay R, Jaakkola TS. Torsional diffusion for molecular conformer generation. In: *ICLR2022 machine learning for drug discovery.* 2022. <https://doi.org/10.48550/ARXIV.2206.01729>.
31. Wu J, Shen T, Lan H, Bian Y, Huang J. SE(3)-equivariant energy-based models for end-to-end protein folding. *bioRxiv.* 2021. <https://doi.org/10.1101/2021.06.06.447297>.
32. Anand N, Achim T. Protein structure and sequence generation with equivariant denoising diffusion probabilistic models. *arXiv:2205.15019.* 2022. <https://doi.org/10.48550/ARXIV.2205.15019>.
33. Trippe BL, Yim J, Tischer D, Baker D, Broderick T, Barzilay R, Jaakkola T. Diffusion probabilistic modeling of protein backbones in 3D for the motif-scaffolding problem. *arXiv:2206.04119.* 2022. <https://doi.org/10.48550/ARXIV.2206.04119>.
34. Qiao Z, Nie W, Vahdat A, Miller TF, Anandkumar A. Dynamic-backbone protein-ligand structure prediction with multiscale generative diffusion models. *arXiv:2209.15171.* 2022. <https://doi.org/10.48550/ARXIV.2209.15171>.
35. Corso G, Stärk H, Jing B, Barzilay R, Jaakkola T. DiffDock: diffusion steps, twists, and turns for molecular docking. *arXiv:2210.01776.* 2022. <https://doi.org/10.48550/ARXIV.2210.01776>.
36. Schneuing A, Du Y, Harris C, Jamasb A, Igashov I, Du W, Blundell T, Lió P, Gomes C, Welling M, Bronstein M, Correia B. Structure-based drug design with equivariant diffusion models. *arXiv:2210.13695.* 2022. <https://doi.org/10.48550/ARXIV.2210.13695>.
37. Liu Z, Li Y, Han L, Li J, Liu J, Zhao Z, Nie W, Liu Y, Wang R. PDB-wide collection of binding data: Current status of the PDBbind database. *Bioinformatics.* 2014;31(3):405–12. <https://doi.org/10.1093/bioinformatics/btu626>.
38. Stärk H, Ganea O-E, Pattanaik L, Barzilay R, Jaakkola T. EquiBind: geometric deep learning for drug binding structure prediction. *Zenodo.* 2022. <https://doi.org/10.5281/zenodo.6408497>.
39. Stärk H, Ganea O-E, Pattanaik L, Barzilay R, Jaakkola T. Equibind: Geometric deep learning for drug binding structure prediction. In: *International conference on machine learning*, 2022;pp. 20503–20521. <https://doi.org/10.48550/ARXIV.2202.05146>.
40. Lu W, Wu Q, Zhang J, Rao J, Li C, Zheng S. TANKBind: Trigonometry-aware neural networks for drug-protein binding structure prediction. *bioRxiv.* 2022. <https://doi.org/10.1101/2022.06.06.495043>.
41. Berman HM. The protein data bank. *Nucleic Acids Res.* 2000;28(1):235–42. <https://doi.org/10.1093/nar/28.1.235>.
42. Kingma DP, Ba J. Adam: A method for stochastic optimization. In: *International conference on learning representations.* 2014. <https://doi.org/10.48550/ARXIV.1412.6980>.
43. Zhang Y. TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res.* 2005;33(7):2302–9. <https://doi.org/10.1093/nar/gki524>.

44. Meller A, Ward M, Borowsky J, Lotthammer JM, Kshirsagar M, Oviedo F, Ferres JL, Bowman GR. Predicting the locations of cryptic pockets from single protein structures using the PocketMiner graph neural network. *bioRxiv*. 2022. <https://doi.org/10.1101/2022.06.28.497399>.
45. Kryshchak A, Schwede T, Topf M, Fidelis K, Moulton J. Critical assessment of methods of protein structure prediction (CASP)–Round XIV. *Proteins: Struct Funct Bioinform*. 2021;89(12):1607–17. <https://doi.org/10.1002/prot.26237>.
46. Evans R, O'Neill M, Pritzel A, Antropova N, Senior A, Green T, Židek A, Bates R, Blackwell S, Yim J, Ronneberger O, Bodenstein S, Zielinski M, Bridgland A, Potapenko A, Cowie A, Tunyasuvunakool K, Jain R, Clancy E, Kohli P, Jumper J, Hassabis D. Protein complex prediction with AlphaFold-Multimer. *bioRxiv*. 2021. <https://doi.org/10.1101/2021.10.04.463034>.
47. Ahdritz G, Bouatta N, Kadyan S, Xia Q, Gerecke W, O'Donnell TJ, Berenberg D, Fisk I, Zanichelli N, Zhang B, et al. OpenFold: Retraining AlphaFold2 yields new insights into its learning mechanisms and capacity for generalization. *bioRxiv*. 2022. <https://doi.org/10.1101/2022.11.20.517210>.
48. DiMaio F, Tyka MD, Baker ML, Chiu W, Baker D. Refinement of protein structures into low-resolution density maps using Rosetta. *J Mol Biol*. 2009;392(1):181–90. <https://doi.org/10.1016/j.jmb.2009.07.008>.
49. Suzek BE, Wang Y, Huang H, McGarvey PB, C.H.W. UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*. 2014;31(6):926–932. <https://doi.org/10.1093/bioinformatics/btu739>.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.

Learn more biomedcentral.com/submissions

