Fine-Tuning DiffDock-L for Allosteric Kinase Docking

Eric Chen,[†,¶] Justin Green,[†,§] and Yingkai Zhang[*,‡,‖]

†*These authors contributed equally to this work.*

‡*Simons Center for Computational Physical Chemistry at New York University, New York, NY 10003, United States*

¶*Department of Chemistry, New York University, New York, NY 10003, United States*

§*Department of Biology, New York University, New York, NY 10003, United States*

E-mail: yingkai.zhang@nyu.edu

Abstract

Allosteric kinase inhibitors are an important modality for overcoming resistance and achieving selectivity, yet most structure-based docking and deep generative models are trained predominantly on orthosteric protein–ligand complexes. As a result, current methods often misplace allosteric kinase ligands into the ATP-binding site and fail to recover the correct binding mode. Here we curate AlloSet, a kinome-wide, time-split dataset of kinase–ligand complexes annotated by binding mode, to systematically evaluate and fine-tune the diffusion-based docking model DiffDock-L for allosteric pose prediction. We explore several fine-tuning strategies, including increased dropout, freezing of torsion parameters with translation/rotation–only fine-tuning, and molecular dynamics (MD)–based supersampling of receptor conformations and ligand poses. The resulting DiffDock-L-Allo model is found to markedly improve pose-recovery metrics for Type III/IV allosteric inhibitors while preserving performance on ATP-site ligands. Binding-mode–resolved evaluations and comparisons with co-folding models such as AlphaFold3 and Boltz-2 highlight how targeted retraining reshapes the generative model's sampling distribution,

offering practical guidance for adapting AI-driven docking to challenging, low-data binding modes in in kinase structure-based drug design.

Introduction

Kinases are an important class of proteins for their multifaceted roles in cellular function from cell division, metabolism, immune response, signal transductions, etc.[1] The human protein kinase family consists of 518 members with distinct functional roles, and there is a wealth of biochemical and structural data available that we can leverage.[2] Kinases function in balance with phosphatases to regulate signaling pathways. Kinases catalyze the transfer of a phosphate group from ATP to substrates, whereas phosphatases remove these phosphate groups, thereby modulating the activity of downstream proteins. The dysregulation of kinases have long been associated with many diseases and cancers, and thus are the target of many drug discovery research campaigns, particularly for antineoplastics.[3,4]

Orthosteric inhibitors, which compete with and sterically block ATP hydrolysis at the catalytic site, are the most common FDA-approved drugs designed to target kinases. There are fewer approved allosteric inhibitors, which inhibit the kinase activity by binding to other sites.[5,6] There has been increasing interest in recent years in the discovery of allosteric inhibitors because kinase catalytic sites tend to be highly evolutionarily conserved, which makes the risk of off-target effects and toxicity substantial for orthosteric inhibitors.[7] Additionally, since cancer can often evolve drug resistance through point mutations, it is desirable to develop therapeutics with diverse binding modalities.[4] Despite these advantages, most allosteric modulators have been discovered serendipitously and there are still challenges in their design.[7,8] These challenges stem from the complexity of modeling the diverse processes involved in protein–allosteric modulator interactions and the difficulty of developing high-throughput assays to search for allosteric binders.[9,10]

Recent trends in computational chemistry research have seen the development of deep learning-based "blind docking" methods for predicting ligand binding poses.[11–13] Blind docking methods

search the receptor for binding pockets in an unrestricted manner. This is in contrast to traditional docking methods that restrict the search space to a particular site.[14] Blind docking algorithms may prove useful for allosteric drug development campaigns since they don't require prior knowledge of the docking site, and allow for post-prediction filtering to remove orthosteric binders.[15] Furthermore, docking methods that robustly sample and predict compounds at various binding sites are necessary for building blind docking methods that are generalizable to a novel target with unknown binding sites. In this work, we focus on binders targeting the kinome because of the clinical relevance of the superfamily and the vast amount of structural data to validate our methods.

Previous work has developed a data-driven platform to benchmark the performance of docking methods within the context of its receptor conformation and ligand binding mode and highlights the challenges that current deep learning models face.[15] We have observed that deep learning-based docking methods predominantly predict ligand binding at the orthosteric site and struggle to sample allosteric sites effectively. An empirical perspective of generative models is that they sample from a learned distribution that approximates the true data distribution. In our case, the training data contains fewer allosteric ligand–kinase complexes than orthosteric ones, and as a result, we anticipate models trained on this data to predominately sample from these high-density regions to predict ligands bound to the orthosteric site. To make generative models useful for predicting allosteric ligands, we must extract signal from this lower-populated data regime. To this aim, we curate a dataset of kinase structures and evaluate a few fine-tuning strategies to improve sampling diversity.

It has become popular in the Large Language Model (LLM) community to take a pretrained model and fine-tune it with a relatively short training cycle on a particular task.[16] This circumvents the extensive computational resources required to train large deep learning models from scratch and allows models to be tailored to particular tasks. We hypothesize that by fine-tuning DiffDock-L on a dataset highly enriched for the types of complexes we are interested in, we can improve our performance metrics. We envision that this strategy can be particularly useful in settings where new data is iteratively introduced such as the Design-Make-Test-Analyze cycle in drug discovery.

Methods

Curation of AlloSet

We curate a kinome-wide structural dataset of kinase binders by taking the union of the Kinase–Ligand Interaction Fingerprints and Structures (KLIFS) database[17,18] and Modi and Dunbrack database[19,20] (obtained on 11/20/24) (**Figure 1**). We rely on KLIFS for their definition of KLIFS residues, a subset of structurally conserved and functionally important residues around the catalytic region selected for consistency across the kinome. We rely on the Modi and Dunbrack database for their heuristic definitions of the ligand binding mode. We briefly describe the heuristics below based on Aurora A Kinase residue numbering.[19,20]

- ATP binding region and hinge residues: 211–213
- Back pocket—αC-helix and partial regions of β4-5 strands, X-DFG and DFG-Asp backbone, and DFG-Phe side chain—residues: 166–193, 196–204, 205–207 and 273–275
- Type 2-only pocket, residues on exposed only in DFG-out structures: 184, 188, 247 and 254

The ligand binding modes are then defined in reference to these residues.

- Type IV: Any small molecule in the asymmetric unit whose minimum distances from the hinge region and αC-helix-Glu(+4) residues are both >6.5 Å.
- Type I ½ front: at least three or more contacts in the back pocket and at least one contact with the N-terminal region of the αC-helix.
- Type I ½ back: at least three or more contacts in the back pocket but no contact with the N-terminal region of the αC-helix.
- Type II: three or more contacts in the back pocket and at least one contact in the Type II-only pocket.
- Type III: minimum distance from the hinge >6 Å and at least three contacts in the back pocket.

- Type I: all the ligands which do not satisfy the above criteria.

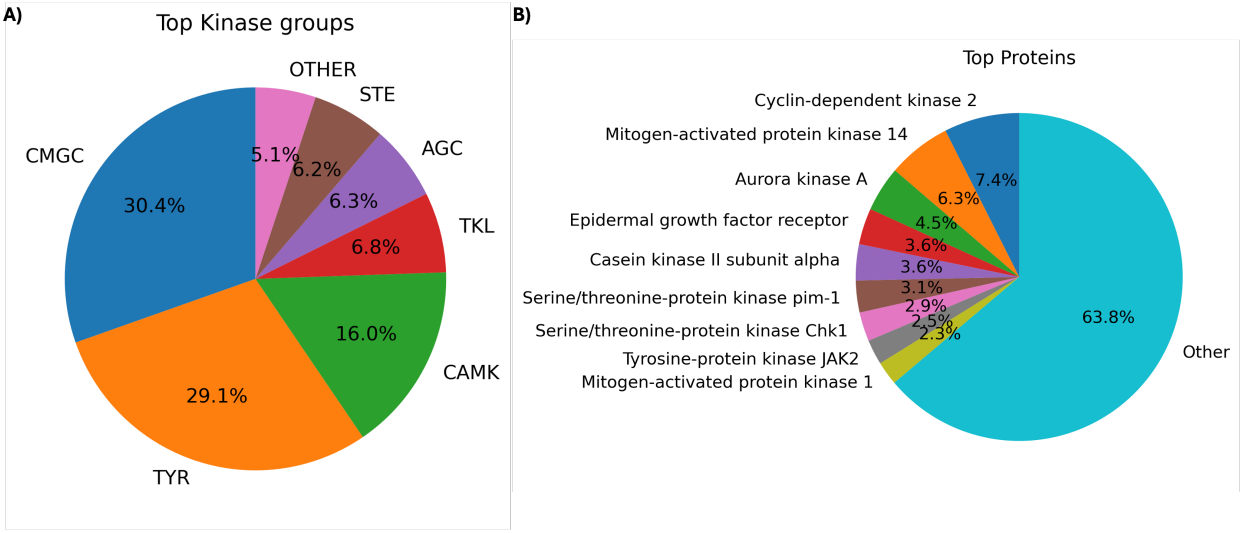


Figure 1: Distribution of AlloSet kinase **A)** group and **B)** name

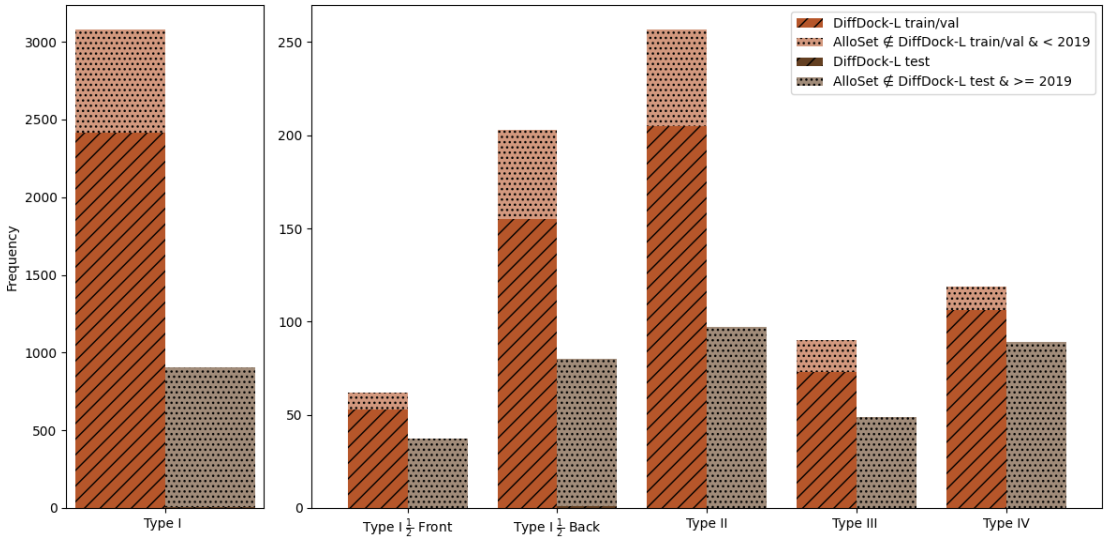


Figure 2: The AlloSet is time-split on 1/1/2019. The binding modes are defined by the Modi and Dunbrack heuristics.

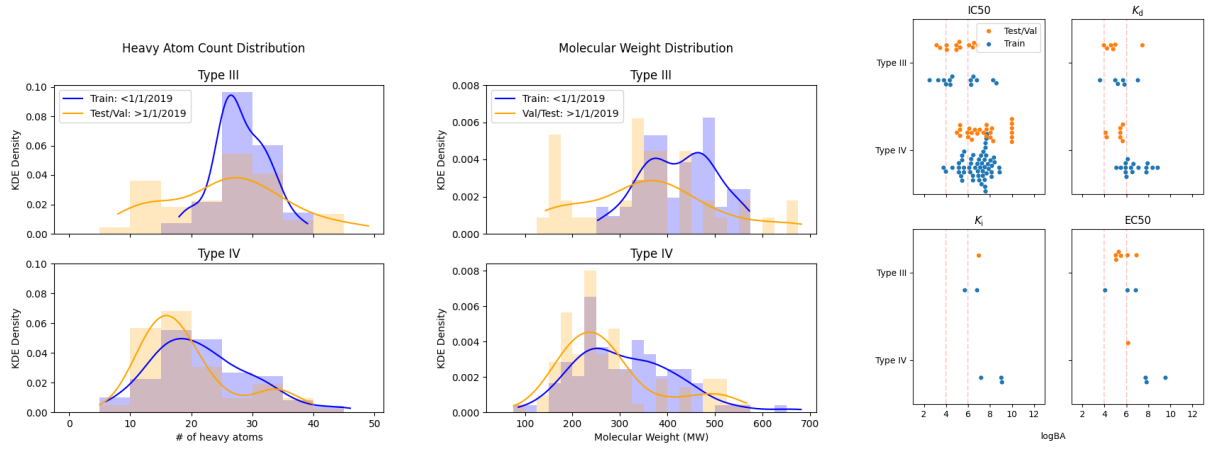
We parse the binders and corresponding monomer chains from the structure. We exclude ions, crystallizing artifacts, and small molecules like glycerol by excluding compounds in the list defined in AlphaFold3 along with some manual curation.[21] We do not remove non-canonical amino acids from the structures. The residues are renumbered according to the UniProt numbering from

Table 1: Breakdown of the AlloSet by binding mode and time-split.

	Time-split date	Type* I	I$\frac{1}{2}$F	I$\frac{1}{2}$B	II	III	IV
AlloSet	<1/1/2019	3082	62	203	257	90	119
	>=1/1/2019	903	37	80	97	49	89
Total		3985	99	283	354	139	208

* F: Front; B: Back

SIFTS.[22] The dataset is split using a 1/1/2019 time-split to follow the convention in Corso et al. where a 2019 time-split was used to train DiffDock-L (**Figure 2, Table 1**)).[23] More details about the training and splitting strategy can be found in **Training and Inference**. Our dataset curation process is able to obtain additional complexes for the training and testing sets beyond the PDBBind dataset, although the dataset is imbalanced in favor of Type I inhibitors. We additionally report the distribution of select molecular properties of the Type III and Type IV ligands and observe that the test split is well represented by the training split (**Figure 3**).



(a) Heavy Atom (b) Molecular Weight (c) Log(binding affinity)

Figure 3: The (a) heavy atom, (b) molecular weight and (c) log-transformed binding affinity distribution of the Type III and Type IV ligands in the training and test splits. A kernel density estimation (KDE) determines the indicated probability density distribution of heavy atom and molecular weights.

Hydrogen Correction

We initially observed inconsistencies with how ligands features, such as bond type, aromaticity, and formal charge, are determined and attributed it to the ambiguities in the ligand pre-processing stage. To address this, we perform a pre-processing step that explicitly defines hydrogen positions in the ligand files. For each ligand in the AlloSet, we use OpenBabel to predict hydrogens at the pH of 7.4 . Then, we use RDKit to check for the proper hydrogenation and charge definitions of imidazole, amino, nitro, tetrazole, and carboxyl groups and correct them if they contain chemical violations.[24,25] Finally, a "corrected" ligand file is produced with explicit hydrogens.

DiffDock

In its initial release, DiffDock achieves state-of-the-art performance of 38% RMSD <2 Å success rate on the PDBBind test set, but its more recent and larger adaptation, DiffDock-L, surpasses it with a 43% success rate.[12,23] The major advancement in DiffDock is its treatment of molecular docking as a diffusion *generative* modeling problem. A diffusion model takes a noisy prior distribution as input and uses a neural network, called the "score model", to iteratively "de-noise" the prior distribution into samples of some target distribution. In this case, the noisy prior distribution is a ligand-protein complex where the ligand is located randomly in space, and the samples generated will places the ligand in a binding pocket. Rather than diffusing in the product space of the 3D coordinates of every atom in the ligand, DiffDock holds bond lengths, angles, and rings rigid, and the diffusion process is defined over the "ligand pose manifold" $\mathscr{M}_c$ (space of all possible ligand poses). The authors postulate that this manifold corresponds directly to the simpler product space $\mathbb{P}$ of the translational (tr; $\mathbb{T}^3$), rotational (rot; $SO(3)$), and torsional (tor; $SO(2)^m$) degrees of freedom of a ligand, where m is the number of possible torsion angles.

$$A_{\mathrm{c}}^{-1} : \mathscr{M}_c \rightarrow \mathbb{P} = \mathbb{T}^3 \times SO(3) \times SO(2)^m \tag{1}$$

Diffusion Models

The core model principle is to train a denoising score matching model, which represents a manifold with a score-function (or the gradient of the log probability density function ($s_\theta(\mathrm{x}) \approx \nabla_x \log p(\mathrm{x})$).[26] During the training process, samples x from the training dataset are made noisy using the traditional forward diffusion process with a stochastic differential equation

$$d\mathrm{x} = \mathrm{f}(\mathrm{x},t)dt + g(t)d\mathrm{w}, t \in (0,T) \tag{2}$$

where w is the Wiener process or Brownian motion, f is the drift function, and $g(t)$ is the diffusion coefficient. When we approach a large T, the ending distribution $p_T(\mathrm{x})$ approaches simple Gaussian noise. This ending noisy distribution is then the prior that we sample from during reverse diffusion. The reverse diffusion process is formalized by the equation

$$d\mathrm{x} = [\mathrm{f}(\mathrm{x}_t,t)dt - g^2(t)\nabla_x \log p_t(\mathrm{x})]dt + g(t)d\bar{\mathrm{w}} \tag{3}$$

where $\nabla_x \log p_t(x)$ is the "score function" of our target probability distribution, and is estimated by the neural network. Thus, during inference of the ligand pose we can sample our target probability distribution using a geodesic random walk with the score as the drift term.[27]

In practice, reverse diffusion during inference amounts to obtaining the relative translation, rotations, and torsion angles updates from a learned distribution that was trained on how to denoise (reverse diffuse) these updates.

Model Architecture

DiffDock uses a message-passing heterogeneous graph architecture that rely on $SE(3)$-equivariant convolutional networks as implemented in E3NN.[28] The score model embeds proteins with Cα as nodes and ESM-2 language model embeddings,[29] and embeds ligand heavy atoms as nodes and physicochemical features. The edges are defined based on a distance cutoffs that depend on

node type and diffusion time. The DiffDock-S and -L models improve the score model architecture compared to DiffDock for greater depth without increasing the runtime, and simplify and balance the confidence model architecture and training.[23]

The E3NN framework is built into the interaction and output layers of the model through tensor product convolutions to ensure equivariance for data/training efficiency and generalization.[28] In other words, E3NN ensure that the samples transform in a predictable way even when the inputs undergo rotational/translational/reflection (SE(3) group) changes.

Confidence Model

Finally, a trained confidence model uses an all-atom representation as input to rank the best binding pose. To train the confidence model, the trained score model generate poses for every training example and labeling each pose if the RMSD < 2 Å. The confidence model is then trained with cross-entropy loss to predict a scalar representing the binary classification of the pose being correct or incorrect.

Training and Inference

While Corso et al. drew both their training and validation set from their pre-2019 time split for training DiffDock-L, we drew our fine-tuning training set from the pre-2019 time split and the validation set from the post-2019 time split to avoid contaminating our validation set with samples that were used in training DiffDock-L (**Figure 2, Table 2**).[23] For testing, we de-duplicate examples with the same UniProt ID and ligand name, and then used the 844 Type I and a stratified random selection of 34 Type III and 62 Type IV crystal structures after the 1/1/2019 time split. The remaining 41 Type III and Type IV structures after the time split not used for testing were used as the validation set.

For validation and testing, we generate 10 complexes for a given receptor-ligand pair, and then use the confidence model to rank them. We track our performance using two main error metrics: the symmetry corrected RMSD between ligand atoms and their ground truth positions ("RMSD"), and

Table 2: Table of the train, validation and test set size used in our experiments, compared to the Boltz-2 and AlphaFold3 test set

		Type*					
	Time-split date	I	$I\frac{1}{2}F$	$I\frac{1}{2}B$	II	III	IV
Fine-tune train	<1/1/2019					86	113
Fine-tune val	>=1/1/2019					15	26
Fine-tune test	>=1/1/2019	844				34	62
Boltz-2/AlphaFold3 test	>9/30/2021	309				8	32

* F: Front; B: Back

the Euclidean distance between the center of the ligand and its ground truth center ("Centroid").[30] We track these metrics both for the top ranked complex ("Top 1"), and the best value for any of the 10 complexes generated ("Any"). We define a prediction as successful in a given metric when the value is below 2 Å.

Sampling Temperature

DiffDock-L has an optional "temperature sampling" procedure that is configured by default. When computing the perturbations of the ligand at each time step (i.e. Δr, ΔR, and $\Delta\theta$), the procedure is

modified for temperature sampling from the original algorithm to the following:

$$\varepsilon_{\text{tr}} \leftarrow e^{\varepsilon_{\text{tr0}} \ln(\sigma_{\text{max tr}}) + (1-\varepsilon_{\text{tr0}}) \ln(\sigma_{\text{min tr}})} \tag{4}$$

$$\lambda_{\text{tr}} \leftarrow \frac{\varepsilon_{\text{tr}} + \sigma_{\text{tr}}}{\varepsilon_{\text{tr}} + \frac{\sigma_{\text{tr}}}{T_{\text{tr}}}} \tag{5}$$

$$\Delta r \leftarrow \Delta \sigma_{\text{tr}}^2 (\lambda_{\text{tr}} + T_0 \frac{\psi_{\text{tr}}}{2}) \alpha + \sqrt{t(1+\psi_{\text{tr}})} z_{\text{tr}} \tag{6}$$

$$\varepsilon_{\text{rot}} \leftarrow e^{\varepsilon_{\text{rot0}} \ln(\sigma_{\text{max rot}}) + (1-\varepsilon_{\text{rot0}}) \ln(\sigma_{\text{min rot}})} \tag{7}$$

$$\lambda_{\text{rot}} \leftarrow \frac{\varepsilon_{\text{rot}} + \sigma_{\text{rot}}}{\varepsilon_{\text{rot}} + \frac{\sigma_{\text{rot}}}{T_{\text{rot}}}} \tag{8}$$

$$\Delta R \leftarrow \Delta \sigma_{\text{rot}}^2 (\lambda_{\text{rot}} + T_0 \frac{\psi_{\text{rot}}}{2}) \beta + \sqrt{t(1+\psi_{\text{rot}})} z_{\text{rot}} \tag{9}$$

$$\varepsilon_{\text{tor}} \leftarrow e^{\varepsilon_{\text{tor0}} \ln(\sigma_{\text{max tor}}) + (1-\varepsilon_{\text{tor0}}) \ln(\sigma_{\text{min tor}})} \tag{10}$$

$$\lambda_{\text{tor}} \leftarrow \frac{\varepsilon_{\text{tor}} + \sigma_{\text{tor}}}{\varepsilon_{\text{tor}} + \frac{\sigma_{\text{tor}}}{T_{\text{tor}}}} \tag{11}$$

$$\Delta \theta \leftarrow \Delta \sigma_{\text{tor}}^2 (\lambda_{\text{tor}} + T_0 \frac{\psi_{\text{tor}}}{2}) \gamma + \sqrt{t(1+\psi_{\text{tor}})} z_{\text{tor}} \tag{12}$$

Where the ε, T, and ψ variables are new configurable values.[23] We hypothesized that this temperature sampling procedure might be ill-suited for our dataset, and experimented with simply disabling it altogether.

Fine-Tuning

From the dataset we collated, we choose the 209 Type III and Type IV complexes that has been uploaded to the PDB before 2019 to use as our training set, and 41 of the post-2019 Type III and Type IV complexes to use as a validation set to track training progress. Using the pretrained DiffDock-L model distributed by Corso et al., we fine-tune the model with this new dataset for 5000 epochs using a learning rate of 0.0001.[23] We monitor our RMSD metrics of the fine tuned model on our validation set every 100 epochs, and saved the model parameters when the rolling average "Any RMSD" metric of the the last 10 validation runs (1000 epochs) were at their best. Progress is monitored using the "Weights and Biases" platform.[31] The various experiments are described in

detail below and summarized in **Table 3**.

Table 3: Comparison of the hyperparameters of the fine-tuning experiments

Run Name	Alpha	Beta	Torsion Loss	Dropout	Dataset Size	Temp Sampling
Baseline	N/A	N/A	N/A	N/A	N/A	True
Baseline No Temp	N/A	N/A	N/A	N/A	N/A	**False**
Fine Tuning	1.0	1.0	0.33	**0.1**	209	True
Dropout	1.0	1.0	0.33	0.5	209	True
MD Supersample	1.0	1.0	0.33	0.5	**627**	True
tr/rot only	**1.25**	**2.25**	**0.0**	0.5	209	True

Dropout

Compared to the 17k complex PDBBind dataset used to pretrain DiffDock-L, our fine tuning training dataset is more than an order of magnitude smaller.[23,32] Because of this discrepancy, we have concerns that our fine tuning will lead to overfitting. One common strategy for addressing overfitting is "dropout", which involves randomly setting some population of neurons to 0 during training.[33] By default, DiffDock-L's training script uses a 10% dropout rate, so we experiment with increasing the dropout rate to 50%.

Molecular Dynamics Supersampling

Similarly to Boltz-2, we also try to combat overfitting by augmenting out dataset with snapshots of molecular dynamics simulations of our experimentally determined structures.[34] For this purpose, we use AMBER 2023 to run 300K, 2ns explicit solvent (TIP3P) simulations of each complex in our training set.[35]

Missing regions in receptor structures are first corrected using the procedure outlined in **S1 Method**. Ligands are hydrogenated using the procedure outlined in **Hydrogen Correction**. Re-

ceptors are protonated using PDB2PQR.[36] Ligand parameterization is done using the GAFF2 force field.[37] Input file preparation were handled by the Antechamber toolkit.[38] Simulations are performed using the FF14SB force field.[39] Energy minimization is first performed with restraints added to receptor and ligand atoms of 2 kcal/mol. Heating is performed from 10K to 300K over 50ps with 2 kcal/mol restraints again added to receptor and ligand atoms. Density equilibration is done at 300K for 50ps with a pressure coupling constant (taup) of 1.0, and the same restraints as the energy minimization and heating steps. Two 100ps equilibration steps are performed before the final MD run: the first with the receptor and ligand restraints reduced to 0.5 kcal/mol, and the second without restraints.

For each simulation, we select 2 frames at random which have an RMSD < 2 Å compared to the ground truth crystal structure, effectively tripling the size of our training dataset.

Translation and Rotation Loss Only

DiffDock-L samples t from a uniform distribution when computing the score-matching loss function for training and then sums together three sub-loss functions: one for translation, one for rotation, and one for torsion scores. DiffDock-L provides weighting parameters for the final loss computation, but by default sets all three weights to 0.33.[12,23] We try a different strategy called tr/rot_only where we train a model to sample t from a β-distribution function with $\alpha = 1.25$ and $\beta = 2.25$ (Equation 13) to bias the early stages of reverse diffusion and set the torsion loss to 0.0.

$$f(x;\alpha,\beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)} \tag{13}$$

where

$$B(\alpha,\beta) = \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}\,dt \tag{14}$$

Co-folding

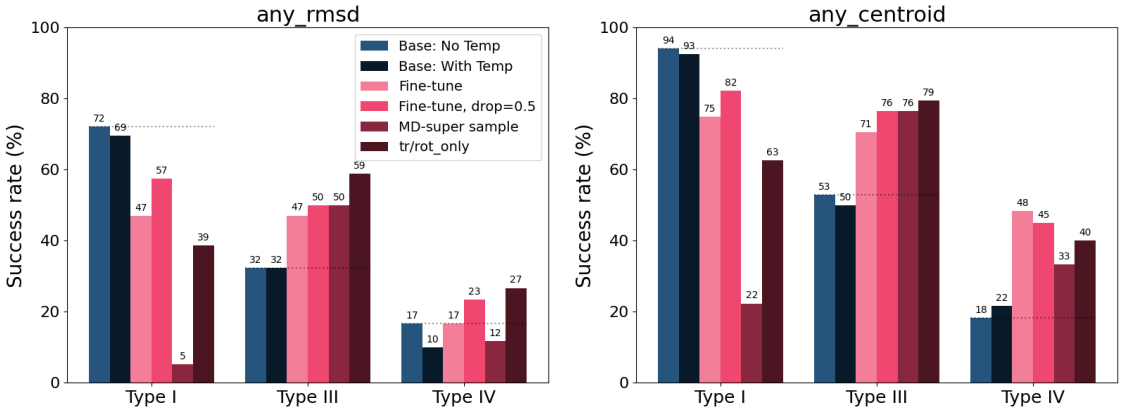
For both AlphaFold3 and Boltz2, we generate 5 samples for each complex using default parameters. We use the Open-structure tools to calculate the protein– and protein–ligand interaction (PLI) local Distance Different Test (lDDT) scores.[40,41] To calculate the RMSD and centroid distances for the co-folding results, we first perform a global sequence alignment with Biopython to obtain consistent residue numbering and then perform an all–Cα superimposition.[42,43] It should be noted that the test set size for the co-folding method is much smaller than the DiffDock-L because AlphaFold3 and Boltz-2 uses a 9/30/2021 time-split while DiffDock-L use a 1/1/2019 time-split (**Table 2**).

Results and Discussion

Fine-tuning DiffDock-L improves docking predictions

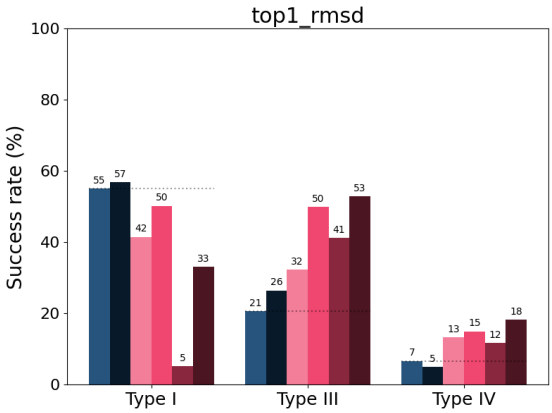
As a baseline, we evaluate the performance of DiffDock-L with and without temperature sampling on the AlloSet test set when predicting 10 samples per complex using only the Type III and Type IV binding poses in our curated dataset before 1/1/19, consistent with the DiffDock-L training set (**Figure 4**). We use two metrics, RMSD and centroid distance between the predicted pose and the reference crystal structure ligand.[30] We denote the fraction of complexes that sampled any pose that satisfy the metric with the "Any" label, and the best confidence-model ranked performance with the "Top1" label. We observe that temperature sampling make little difference in our success metrics, suggesting this may be an unnecessary step for some datasets. Notably, we observe a significant drop off in performance when docking Type III and Type IV compared to the Type I ligands. The centroid-distance results suggest that the vast difference in performance occurs because the docking model is unable to frequently sample the allosteric poses. These results are consistent with our observations in our previous work.[15]

In all cases, fine-tuning is able to improve the prediction of the Type III and IV binding pose compared to the baseline with varying magnitude of performance decreases in prediction of the

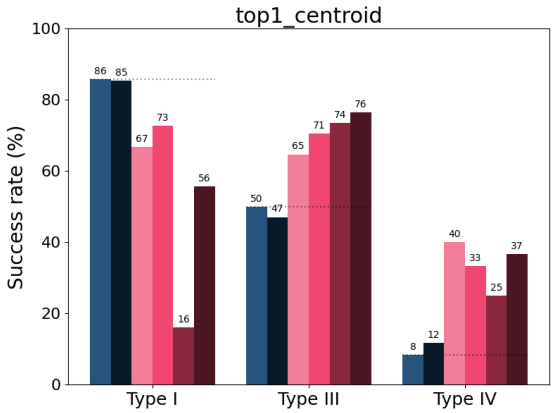


(a) Any RMSD < 2 Å

(b) Any Centroid < 2 Å



(c) Top 1 RMSD < 2 Å



(d) Top 1 Centroid < 2 Å

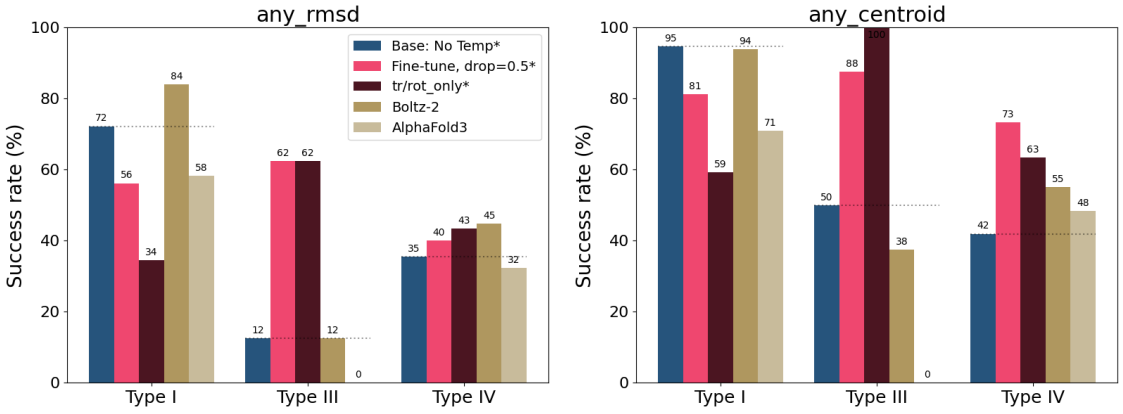
Figure 4: Plots showing the fraction of DiffDock-L predicted complexes in the test set that satisfy the requirements (a) if the top 1 confidence poses result in RMSD < 2 Å or (b) centroid distance < 2 Å and (c) if any of the 10 sampled poses that result in RMSD < 2 Å or (d) centroid distance < 2 Å. The blue bars indicate that the baseline DiffDock-L models with and without temperature sampling. The red bars indicate the various fine-tuning strategies such as with only the allosteric training set with drop=0.1 or 0.5, MD-super sampling, and updating only the translation and rotation heads and freezing the torsion head (tr/rot_only). The dotted line indicates the baseline performance without temperature sampling.

Type I binding pose. The fine-tune drop=0.5 and tr/rot_only strategies roughly double the baseline performance, with tr/rot_only performing best in predicting the Type III and IV binding mode with RMSD < 2 Å in any of the 10 samples (**Figure 4a**) and with fine-tune drop=0.5 performing best in predicting the Type IV binding mode with centroid < 2 Å (**Figure 4b**). The intuition behind the DiffDock-L model is that that the early stages of reverse diffusion explores the binding sites on the receptor through the translation and rotation updates, and the later stages position the ligand in the binding through rotation and torsional updates. The slight improvement we see from the tr/rot_only strategy lends support to the hypothesis that focusing our fine tuning efforts on only translation and rotation could improve the frequency by which DiffDock-L selects the correct binding pocket.

We find that all fine-tuned models lead to a decrease in performance on Type I ligands. Of the methods tested, fine tuning with increased dropout results in the lowest reduction Type I performance. Including Type I validation set performance as another early stopping criteria can counter this trade-off that we observe. Interestingly, the largest decrease in performance for Type I poses occurs when we super-sample the training set with crystal poses. We speculate that a large increase of the number of new training data-points leads to catastrophic forgetting. For Type IV binders, we do not see a strong improvement in RMSD but observe encouraging improvements in centroid performance. Overall, centroid performance is consistently higher than RMSD indicating that the models are finding the correct site, but not predicting the perfect pose.

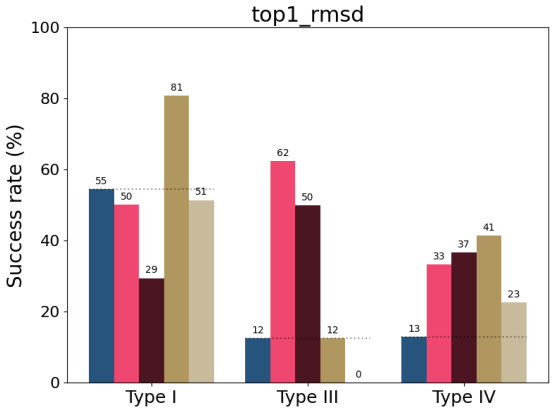
Co-folding methods still have challenges

We also evaluate Boltz-2 while providing a receptor template, and Boltz-2 and AlphaFold3 to jointly predict the protein–ligand binding mode on our dataset (**Figure 5, 6**). It should be noted that our test set size is much smaller because AlphaFold3 and Boltz-2 use a 9/30/2021 time-split while DiffDock-L uses a 1/1/2019 time-split (**Table 2**). To assess each complex, we supply the protein sequence and hydrogen corrected SMILES as a ligand, and evaluate the protein– and PLI–lDDT scores. On this set, we observe that Boltz-2 and AlphaFold3 lead to a drop off in performance similar to DiffDock-L baseline when predicting allosteric poses compared to orthosteric poses in

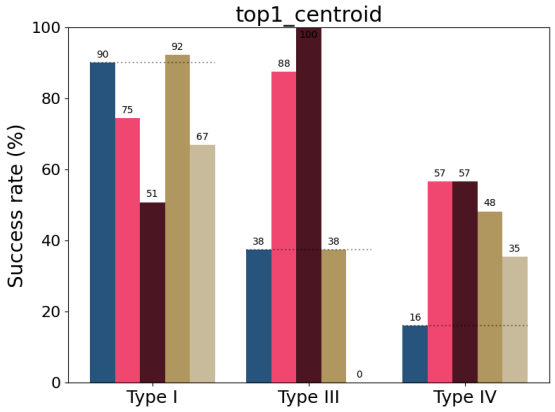


(a) Any RMSD < 2 Å

(b) Any Centroid < 2 Å

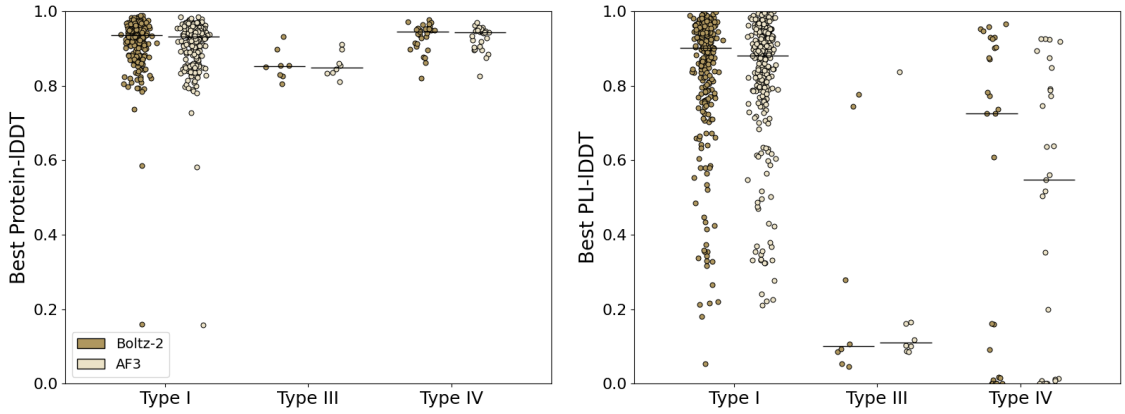


(c) Top 1 RMSD < 2 Å



(d) Top 1 Centroid < 2 Å

Figure 5: Plots showing the fraction of the Boltz-2 and AlphaFold3 co-folding methods that satisfy the requirements if (a) the top 1 ranked poses result in RMSD < 2 Å or (b) centroid distance < 2 Å and (c) if any of the 10 sampled poses that result in RMSD < 2 Å or (d) centroid distance < 2 Å. The dotted line indicates the performance of DiffDock-L baseline without temperature. For the co-folding methods, we apply an all-Cα superimposition prior to calculating the RMSD and centroid distance. *Note that the DiffDock-L and fine-tuned results shown here reflect the 9/30/2021 time-split.



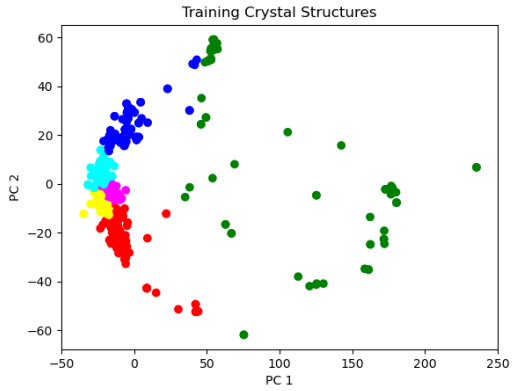
(a) Protein–lDDT (b) PLI–lDDT

Figure 6: Strip plots of the best global (a) protein and (b) PLI–lDDT scores of AlphaFold3 and Boltz-2 on the test set across 5 samples. The line indicates the median performance for the particular binding mode.

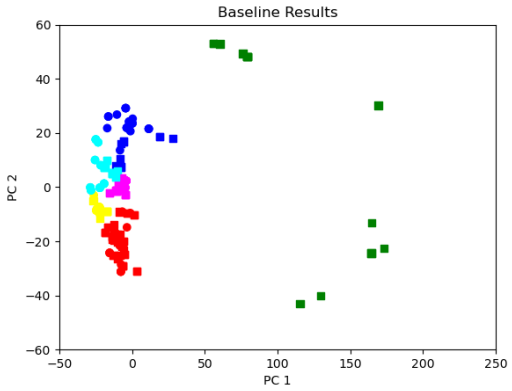
the RMSD, centroid and PLI–lDDT scores despite high protein–lDDT (>0.8) (**Figure 5, 6**). Overall, Boltz-2 performs comparably with and without a template structure, and better than AlphaFold3 across all ligand binding modes. DiffDock-L baseline performs worse than the Boltz-2 on the Type I and Type IV binding modes, but the fine-tuned models has superior performance across Type III binding modes.

Fine Tuning Improves Pocket Selection

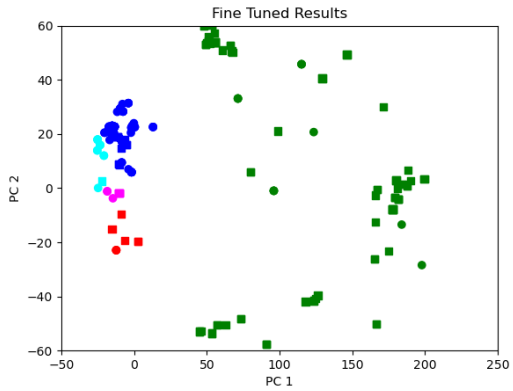
The primary failure mode is that the baseline DiffDock-L model often docks allosteric ligands to the orthosteric pocket. To visualize our observation, we select 84 "KLIFS residues" that are highly conserved among kinases and measure the minimum residue–ligand distance vector.[15,18] This creates a proxy for the protein–ligand binding location for the entire kinase dataset. We then use PCA to reduce the dimensionality of our representation and determine the centroids of each complex type using the labeled ground truth crystal structures (**Figure 7a**). We project the top-1 predicted structures into the PCA vector space, and use the Euclidean distance in the PCA vector space between the pose and the complex type centroids to classify the pose. The label of a particular structure is set to that of the nearest complex type centroid. Despite the fact that the test dataset



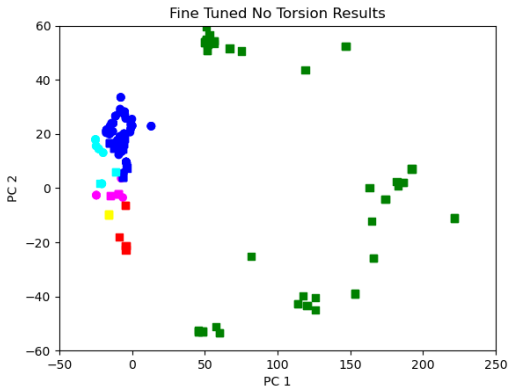
(a) Crystal ligand pose PCA



(b) Baseline results



(c) Fine-tuned results



(d) Tr/rot_only results

Figure 7: Ligand binding location representation space colored by pocket. (a) The PCA of all the crystal ligand pose structures. (b-d) The top-1 predicted poses of each model version is projected onto the crystal ligand pose PCA. For all subfigures, the X axis is the first principle component (explained variance ratio 0.67), and the Y axis is the second principle component (explained variance ratio 0.13). Dots indicate that the ligand comes from a Type III complex, while squares indicate a Type IV complex. The color of each data point indicate which pocket the ligand was actually docked into. Color key: Red=Type I, Yellow=Type 1.5 Back, Magenta=Type 1.5 Front, Cyan=Type II, Dark blue=Type III, Green=Type IV. Ideally for Figures (b-d), all dots should be blue, all squares should be green, and there should be no cyan, magenta, yellow, or red data points of any kind.

only contains Type III and Type IV complexes, **Figure 7b** shows a large number of cyan, magenta, yellow, and red data points. This indicates that the baseline DiffDock-L model is frequently docking Type III ligands in Type I, Type 1.5, and Type II pockets. In contrast, fine tuning, both with (**Figure 7c**) and without torsion loss (**Figure 7d**), results in far fewer cyan, magenta, yellow, and red data points, and far more blue dots and green squares. This indicates that our fine-tuned models are less frequently docking Type III and Type IV ligands inappropriately into the orthosteric binding pockets.

Conclusion

A key challenge in generative modeling for protein–ligand interactions is determining whether a model trained on known interactions can generalize to predict novel binding poses of distinct compounds at different sites or proteins. Commercial value from drug discovery comes from the ability to target novel proteins and expand to novel modalities or chemical space not previously patented or studied. Some researchers suggest that DiffDock-L excels only on proteins represented in the training set.[44] Other work has discussed the idea that deep learning co-folding models are overfit to specific protein families and that the success rate is dictated by the similarity of structures to the training set.[45,46] While the proteins in our test set are certainly well-represented in training set, we ask a distinct question on one particular protein family, can we shift the trained distribution of a generative model to predicted binding poses at a binding pockets underrepresented in the training data?

Answering this question first begins with developing proper benchmarks that splits the data into distinct training and test sets.[23,46–48] In this work, we curate a comprehensive dataset from KLIFS and the Modi and Dunbrack binding mode nomenclature, and use a time split.[19] When we evaluate the performance of these generative models on our test set, we observe that the performance of both DiffDock-L and co-folding models drop off when predicting allosteric ligands compared to orthosteric ligands. This observation with co-folding models is also consistent amongst broader

orthosteric and allosteric datasets.[49,50] Leveraging fine tuning tactics on co-folding models will likely yield similar improvements in performance on tailored datasets such as for allostery. Such strategies are being explored at scale in industry through federated learning of OpenFold3.[21,51,52]

Once a model is developed to sample diverse binding sites, the next step is to rank and discriminate between predicted binding modes. DiffDock-L introduces a confidence model which ranks the protein–ligand binding pose. Even if you sample the right binding site, a scoring function that is able to discriminate between the correct and false poses is needed. To this aim, decoy sets have been used to assess scoring functions.[53–56] This will require a computationally generated set of allosteric decoys (allosteric ligands in the orthosteric site) to assess how robust this confidence model is at ranking binding modes. This approach has utility outside of blind docking. For example, pocket-restricted docking applied to multiple sites of a target protein also require a subsequent ranking function to distinguish the poses in the correct site from the other sites. Furthermore, a target-specific classifier model can be trained on this data for greater screening performance.[57]

In our work, we perform "re-docking" to the original crystal structure. It will also be important to assess our docking model prospectively in a high-throughput screening setting.[58] This adds another layer of complexity where it is not only important to dock and filter ligands appropriately, but also to score and rank binders over non-binders. We can gather the screening data in the literature for a retrospective screen and determine the hit-rate of our fine-tuned model.[59–61] After using our fine-tuned model to dock a set of ligands, we can then use scoring function models that have been particularly tuned for the screening task to rank the binding poses.[57,62] Ideally, the blind docking model will rely on learned features to classify the allosteric and orthosteric ligands and filtering step can remove ligands more likely to be orthosteric. A model that is considered overfit to orthosteric binders for a particular protein class is a beneficial attribute as a negative class filter. A proper test will contrast the performance of this blind docking approach with the performance of restricted search space docking on both orthosteric and allosteric sites.

In this work, we use DiffDock-L with the assumption that the receptor is rigid. Our previous work describe that only one particular conformation allowed allosteric docking success, primarily

because it was a ligand-bound at the allosteric site.[15] AlphaFold2 structures can also be useful input receptors. This may require methods to encourage conformational diversity such as AlphaFold2-based methods that sub-sample the multiple sequence alignment[63–67] and those that produce distributions of conformations[68–71] or reveal cryptic pockets to receptor conformations primed for allosteric binding[72–74] can be useful for determining input receptor structures that will be useful for docking allosteric ligands. A co-folding modeled tuned to produce diverse receptor conformations will circumvent the need to predict receptor structures ahead of time as well.

Next steps can include evaluating the diversity[75–77] and physical validity of the model predictions.[78,79] Strategies like representation space probing can give qualitative interpretations of whether or not a deep learning model can be used for generalization tasks.[80] Advancements can include creative architectures and training strategies. Concepts from the continual learning field can be used to ensure that a deep learning model retains the capability to adapt without being retrained.[81] Integrating first-principles or physics-based intuition into a machine learning model can guide performance in unexplored representation spaces.[82,83] Lastly, elaborate collection of multi-modal training data through high-throughput non-competitive screens or utilization of pre-trained models in translatable knowledge spaces can provide additional information for a model to learn from.[59,84–86]

Data and Software Availability

All code is available at `https://github.com/electrojustin/DiffDock-Fine-Tune`. Datasets are available at `doi:10.5281/zenodo.17373044`

Supporting Information

S1 Method. Conformation-based loop modeling To prepare receptors for the MD simulations use a template-based approach, Modeller, to model missing residues segments from similar protein conformations to the structure being remodelled.

Acknowledgements

Y.Z. acknowledges support from the U.S. National Institutes of Health (NIH) (R35-GM127040). E.A.C. is partially supported by a graduate fellowship from the Simons Center of Computational Physical Chemistry (SCCPC) at NYU. E.A.C thank Bruno Correira, Gabriele Corso, Paula Linh Kramer, Mark Maupin, Eva Nittinger, Qi Ouyang, Steven Pak, Catarina Santos, Dina Sharon, Kunyang (Oliver) Sun, Pat Walters for insightful conversations which helped refine elements of this work.

References

(1) Roskoski, R. A historical overview of protein kinases and their targeted small molecule inhibitors. *Pharm. Res.* **2015**, *100*, 1–23.

(2) Manning, G.; Whyte, D. B.; Martinez, R.; Hunter, T.; Sudarsanam, S. The Protein Kinase Complement of the Human Genome. *Sci.* **2002**, *298*, 1912–1934.

(3) Noble, M. E. M.; Endicott, J. A.; Johnson, L. N. Protein Kinase Inhibitors: Insights into Drug Design from Structure. *Sci.* **2004**, *303*, 1800–1805.

(4) Lu, X.; Smaill, J. B.; Ding, K. New Promise and Opportunities for Allosteric Kinase Inhibitors. *Angewandte Chemie Int. Ed.* **2020**, *59*, 13764–13776.

(5) Pan, Y.; Mader, M. M. Principles of Kinase Allosteric Inhibition and Pocket Validation. *J. of Med. Chem.* **2022**, *65*, 5288–5299.

(6) Laufer, S.; Bajorath, J. New Horizons in Drug Discovery - Understanding and Advancing Different Types of Kinase Inhibitors: Seven Years in Kinase Inhibitor Research with Impressive Achievements and New Future Prospects. *J. of Med. Chem.* **2022**, *65*, 891–892.

(7) Lu, S.; Shen, Q.; Zhang, J. Allosteric Methods and Their Applications: Facilitating the

Discovery of Allosteric Drugs and the Investigation of Allosteric Mechanisms. *Acc. of Chem. Res.* **2019**, *52*, 492–500.

(8) Hardy, J. A.; Wells, J. A. Searching for new allosteric sites in enzymes. *Curr. Opin. in Struct. Biology* **2004**, *14*, 706–715.

(9) Nussinov, R.; Zhang, M.; Liu, Y.; Jang, H. AlphaFold, allosteric, and orthosteric drug discovery: Ways forward. *Drug Discov. Today* **2023**, *28*, 103551.

(10) Wenthur, C. J.; Gentry, P. R.; Mathews, T. P.; Lindsley, C. W. Drugs for Allosteric Sites on Receptors. *Annu. Rev. of Pharm. and Toxicol.* **2014**, *54*, 165–184.

(11) 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**, 2022.06.06.495043.

(12) Corso, G.; Stärk, H.; Jing, B.; Barzilay, R.; Jaakkola, T. DiffDock: Diffusion Steps, Twists, and Turns for Molecular Docking. *arXiv e-prints* **2022**, arXiv:2210.01776.

(13) Morehead, A.; Giri, N.; Liu, J.; Neupane, P.; Cheng, J. Deep Learning for Protein-Ligand Docking: Are We There Yet? 2025; `http://arxiv.org/abs/2405.14108`.

(14) Yang, C.; Chen, E. A.; Zhang, Y. Protein–Ligand Docking in the Machine-Learning Era. *Mol.* **2022**, *27*, 4568.

(15) Chen, E. A.; Zhang, Y. Can Deep Learning Blind Docking Methods be Used to Predict Allosteric Compounds? *J. of Chem. Inf. Model.* **2025**, *65*, 3737–3748.

(16) Dodge, J.; Ilharco, G.; Schwartz, R.; Farhadi, A.; Hajishirzi, H.; Smith, N. Fine-Tuning Pretrained Language Models: Weight Initializations, Data Orders, and Early Stopping. 2020; `http://arxiv.org/abs/2002.06305`.

(17) van Linden, O. P. J.; Kooistra, A. J.; Leurs, R.; de Esch, I. J. P.; de Graaf, C. KLIFS: A Knowledge-Based Structural Database To Navigate Kinase–Ligand Interaction Space. *J. of Med. Chem.* **2014**, *57*, 249–277.

(18) Kanev, G. K.; de Graaf, C.; Westerman, B. A.; de Esch, I. J. P.; Kooistra, A. J. KLIFS: an overhaul after the first 5 years of supporting kinase research. *Nucl. Acids Res.* **2020**, *49*, D562–D569.

(19) Modi, V.; Dunbrack, R. L. Defining a new nomenclature for the structures of active and inactive kinases. *Proc. of the National Acad. of Sci.* **2019**, *116*, 6818–6827.

(20) Modi, V.; Dunbrack, R. L., Jr Kincore: a web resource for structural classification of protein kinases and their inhibitors. *Nucl. Acids Res.* **2022**, *50*, D654–D664.

(21) Abramson, J. et al. Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nat.* **2024**, 1–3.

(22) Dana, J. M.; Gutmanas, A.; Tyagi, N.; Qi, G.; O'Donovan, C.; Martin, M.; Velankar, S. SIFTS: updated Structure Integration with Function, Taxonomy and Sequences resource allows 40-fold increase in coverage of structure-based annotations for proteins. *Nucl. Acids Res.* **2019**, *47*, D482–D489.

(23) Corso, G.; Deng, A.; Fry, B.; Polizzi, N.; Barzilay, R.; Jaakkola, T. Deep Confident Steps to New Pockets: Strategies for Docking Generalization. 2024; `http://arxiv.org/abs/2402.18396`.

(24) O'Boyle, N. M.; Banck, M.; James, C. A.; Morley, C.; Vandermeersch, T.; Hutchison, G. R. Open Babel: An open chemical toolbox. *J. of Cheminformatics* **2011**, *3*, 33.

(25) Landrum, G. et al. rdkit/rdkit: 2020_03_1 (Q1 2020) Release. 2020; `https://zenodo.org/record/3732262`.

(26) Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; Poole, B. Score-Based Generative Modeling through Stochastic Differential Equations. *arXiv.org* **2021**,

(27) Bortoli, V. D.; Mathieu, E.; Hutchinson, M.; Thornton, J.; Teh, Y. W.; Doucet, A. Riemannian Score-Based Generative Modelling. 2022; `http://arxiv.org/abs/2202.02763`.

(28) Geiger, M.; Smidt, T. e3nn: Euclidean Neural Networks. 2022; http://arxiv.org/abs/2207.09453.

(29) 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. *Sci.* **2023**, *379*, 1123–1130.

(30) Meli, R.; Biggin, P. C. spyrmsd: symmetry-corrected RMSD calculations in Python. *J. of Cheminformatics* **2020**, *12*, 1–7.

(31) Biewald, L. Experiment Tracking with Weights and Biases. 2020; https://www.wandb.com/.

(32) 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. *Bioinform.* **2015**, *31*, 405–412.

(33) Hinton, G. E.; Srivastava, N.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. R. Improving neural networks by preventing co-adaptation of feature detectors. 2012; http://arxiv.org/abs/1207.0580.

(34) Passaro, S.; Corso, G.; Wohlwend, J.; Reveiz, M.; Thaler, S.; Somnath, V. R.; Getz, N.; Portnoi, T.; Roy, J.; Stark, H.; Kwabi-Addo, D.; Beaini, D.; Jaakkola, T.; Barzilay, R. Boltz-2: Towards Accurate and Efficient Binding Affinity Prediction. **2025**,

(35) Case, D. et al. Amber 2023. 2023.

(36) Jurrus, E. et al. Improvements to the APBS biomolecular solvation software suite. *Protein Sci.* **2018**, *27*, 112–128.

(37) Wang, J.; Wolf, R. M.; Caldwell, J. W.; Kollman, P. A.; Case, D. A. Development and testing of a general amber force field. *J. of Comput. Chem.* **2004**, *25*, 1157–1174.

(38) Wang, J.; Wang, W.; Kollman, P. A.; Case, D. A. Automatic atom type and bond type perception in molecular mechanical calculations. *J. of Mol. Graph. and Model.* **2006**, *25*, 247–260.

(39) Maier, J. A.; Martinez, C.; Kasavajhala, K.; Wickstrom, L.; Hauser, K. E.; Simmerling, C. ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB. *J. of Chem. Theory and Comput.* **2015**, *11*, 3696–3713.

(40) Mariani, V.; Biasini, M.; Barbato, A.; Schwede, T. lDDT: a local superposition-free score for comparing protein structures and models using distance difference tests. *Bioinform.* **2013**, *29*, 2722–2728.

(41) Biasini, M.; Schmidt, T.; Bienert, S.; Mariani, V.; Studer, G.; Haas, J.; Johner, N.; Schenk, A. D.; Philippsen, A.; Schwede, T. OpenStructure: an integrated software framework for computational structural biology. *Acta Cryst. Sect. D: Biological Cryst.* **2013**, *69*, 701–709.

(42) Cock, P. J. A.; Antao, T.; Chang, J. T.; Chapman, B. A.; Cox, C. J.; Dalke, A.; Friedberg, I.; Hamelryck, T.; Kauff, F.; Wilczynski, B.; de Hoon, M. J. L. Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinform.* **2009**, *25*, 1422–1423.

(43) Durbin, R.; Eddy, S. R.; Krogh, A.; Mitchison, G. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, 1st ed.; Cambridge University Press, 1998.

(44) Jain, A. N.; Cleves, A. E.; Walters, W. P. Deep-Learning Based Docking Methods: Fair Comparisons to Conventional Docking Workflows. 2024; `http://arxiv.org/abs/2412.02889`.

(45) Masters, M. R.; Mahmoud, A. H.; Lill, M. A. Do Deep Learning Models for Co-Folding Learn the Physics of Protein-Ligand Interactions? 2024; `https://www.biorxiv.org/content/10.1101/2024.06.03.597219v1`.

(46) Škrinjar, P.; Eberhardt, J.; Durairaj, J.; Schwede, T. Have protein-ligand co-folding methods moved beyond memorisation? 2025; https://www.biorxiv.org/content/10.1101/2025.02.03.636309v2.

(47) Durairaj, J. et al. PLINDER: The protein-ligand interactions dataset and evaluation resource. 2024; https://www.biorxiv.org/content/10.1101/2024.07.17.603955v3.

(48) Kramer, C.; Chodera, J.; Damm-Ganamet, K. L.; Gilson, M. K.; Günther, J.; Lessel, U.; Lewis, R. A.; Mobley, D.; Nittinger, E.; Pecina, A.; Schapira, M.; Walters, W. P. The Need for Continuing Blinded Pose- and Activity Prediction Benchmarks. *J. of Chem. Inf. Model.* **2025**, *65*, 2180–2190.

(49) Nittinger, E.; Yoluk, O.; Tibo, A.; Olanders, G.; Tyrchan, C. Co-folding, the future of docking prediction of allosteric and orthosteric ligands. *Artif. Intell. in the Life Sci.* **2025**, *8*, 100136.

(50) Olanders, G.; Testa, G.; Tibo, A.; Nittinger, E.; Tyrchan, C. Challenge for Deep Learning: Protein Structure Prediction of Ligand-Induced Conformational Changes at Allosteric and Orthosteric Sites. *J. Chem. Inf. Model.* **2024**, *64*, 8481–8494.

(51) The OpenFold3 Team OpenFold3-preview. 2025; https://github.com/aqlaboratory/openfold-3.

(52) ApherisFold. https://www.apheris.com/apherisfold.

(53) Stein, R. M.; Yang, Y.; Balius, T. E.; O'Meara, M. J.; Lyu, J.; Young, J.; Tang, K.; Shoichet, B. K.; Irwin, J. J. Property-Unmatched Decoys in Docking Benchmarks. *J. of Chem. Inf. Model.* **2021**, *61*, 699–714.

(54) Su, M.; Yang, Q.; Du, Y.; Feng, G.; Liu, Z.; Li, Y.; Wang, R. Comparative Assessment of Scoring Functions: The CASF-2016 Update. *J. of Chem. Inf. Model.* **2019**, *59*, 895–913.

(55) Ibrahim, T. M.; Bauer, M. R.; Boeckler, F. M. Applying DEKOIS 2.0 in structure-based

virtual screening to probe the impact of preparation procedures and score normalization. *J. of Cheminformatics* **2015**, *7*, 21.

(56) Mysinger, M. M.; Carchia, M.; Irwin, J. J.; Shoichet, B. K. Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and Decoys for Better Benchmarking. *J. of Med. Chem.* **2012**, *55*, 6582–6594.

(57) Xia, S.; Gu, Y.; Zhang, Y. Normalized Protein–Ligand Distance Likelihood Score for End-to-End Blind Docking and Virtual Screening. *J. of Chem. Inf. Model.* **2025**, *65*, 1101–1114.

(58) Lyu, J. et al. AlphaFold2 structures guide prospective ligand discovery. *Sci.* **2024**, *384*, eadn6354.

(59) Martin, M. P.; Alam, R.; Betzi, S.; Ingles, D. J.; Zhu, J.-Y.; Schönbrunn, E. A Novel Approach to the Discovery of Small-Molecule Ligands of CDK2. *ChemBioChem* **2012**, *13*, 2128–2136.

(60) Rastelli, G.; Anighoro, A.; Chripkova, M.; Carrassa, L.; Broggini, M. Structure-based discovery of the first allosteric inhibitors of cyclin-dependent kinase 2. *Cell Cycle* **2014**, *13*, 2296–2305.

(61) Faber, E. B. et al. Development of allosteric and selective CDK2 inhibitors for contraception with negative cooperativity to cyclin binding. *Nat. Commun.* **2023**, *14*, 3213.

(62) Shen, C.; Zhang, X.; Deng, Y.; Gao, J.; Wang, D.; Xu, L.; Pan, P.; Hou, T.; Kang, Y. Boosting Protein–Ligand Binding Pose Prediction and Virtual Screening Based on Residue–Atom Distance Likelihood Potential and Graph Transformer. *J. of Med. Chem.* **2022**, *65*, 10691–10706.

(63) Stein, R. A.; Mchaourab, H. S. SPEACH_AF: Sampling protein ensembles and conformational heterogeneity with Alphafold2. *PLOS Comput. Biology* **2022**, *18*, e1010483.

(64) del Alamo, D.; Sala, D.; Mchaourab, H. S.; Meiler, J. Sampling alternative conformational states of transporters and receptors with AlphaFold2. *eLife* **2022**, *11*, e75751.

(65) Monteiro da Silva, G.; Cui, J. Y.; Dalgarno, D. C.; Lisi, G. P.; Rubenstein, B. M. High-throughput prediction of protein conformational distributions with subsampled AlphaFold2. *Nat. Commun.* **2024**, *15*, 2464.

(66) Sala, D.; Hildebrand, P. W.; Meiler, J. Biasing AlphaFold2 to predict GPCRs and kinases with user-defined functional or structural properties. *Frontiers in Mol. Biosci.* **2023**, *10*.

(67) Al-Masri, C.; Trozzi, F.; Lin, S.-H.; Tran, O.; Sahni, N.; Patek, M.; Cichonska, A.; Ravikumar, B.; Rahman, R. Investigating the conformational landscape of AlphaFold2-predicted protein kinase structures. *Bioinform. Adv.* **2023**, *3*, vbad129.

(68) Zheng, S. et al. Predicting equilibrium distributions for molecular systems with deep learning. *Nat. Mach. Intell.* **2024**, *6*, 558–567.

(69) Jing, B.; Berger, B.; Jaakkola, T. AlphaFold Meets Flow Matching for Generating Protein Ensembles. 2024; `http://arxiv.org/abs/2402.04845`.

(70) Pang, Y. T.; Kuo, K. M.; Yang, L.; Gumbart, J. C. DeepPath: Overcoming data scarcity for protein transition pathway prediction using physics-based deep learning. 2025; `https://www.biorxiv.org/content/10.1101/2025.02.27.640693v1`.

(71) Bhakat, S.; Vats, S.; Mardt, A.; Degterev, A. Generalizable Protein Dynamics in Kinases: Physics is the key. 2025; `https://www.biorxiv.org/content/10.1101/2025.03.06.641878v2`.

(72) Meller, A.; Ward, M.; Borowsky, J.; Kshirsagar, M.; Lotthammer, J. M.; Oviedo, F.; Ferres, J. L.; Bowman, G. R. Predicting locations of cryptic pockets from single protein structures using the PocketMiner graph neural network. *Nat. Commun.* **2023**, *14*, 1177.

(73) Vats, S.; Bobrovs, R.; Söderhjelm, P.; Bhakat, S. AlphaFold-SFA: Accelerated sampling of cryptic pocket opening, protein-ligand binding and allostery by AlphaFold, slow feature analysis and metadynamics. *PLOS ONE* **2024**, *19*, e0307226.

(74) Meller, A.; Bhakat, S.; Solieva, S.; Bowman, G. R. Accelerating Cryptic Pocket Discovery Using AlphaFold. *J. of Chem. Theory and Comput.* **2023**, *19*, 4355–4363.

(75) Sehwag, V.; Hazirbas, C.; Gordo, A.; Ozgenel, F.; Ferrer, C. C. Generating High Fidelity Data from Low-density Regions using Diffusion Models. 2022; `http://arxiv.org/abs/2203.17260`.

(76) Corso, G.; Xu, Y.; de Bortoli, V.; Barzilay, R.; Jaakkola, T. Particle Guidance: non-I.I.D. Diverse Sampling with Diffusion Models. 2023; `http://arxiv.org/abs/2310.13102`.

(77) Tossou, P.; Wognum, C.; Craig, M.; Mary, H.; Noutahi, E. Real-World Molecular Out-Of-Distribution: Specification and Investigation. 2023; `https://chemrxiv.org/engage/chemrxiv/article-details/64c012a1b053dad33ae21932`.

(78) Buttenschoen, M.; Morris, G. M.; Deane, C. M. PoseBusters: AI-based docking methods fail to generate physically valid poses or generalise to novel sequences. 2023; `http://arxiv.org/abs/2308.05777`.

(79) Masters, M. R.; Mahmoud, A. H.; Lill, M. A. Investigating whether deep learning models for co-folding learn the physics of protein-ligand interactions. *Nat. Commun.* **2025**, *16*, 8854.

(80) Wan, Y.; Wu, J.; Hou, T.; Hsieh, C.-Y.; Jia, X. Multi-channel learning for integrating structural hierarchies into context-dependent molecular representation. *Nat. Commun.* **2025**, *16*, 413.

(81) Dohare, S.; Hernandez-Garcia, J. F.; Lan, Q.; Rahman, P.; Mahmood, A. R.; Sutton, R. S. Loss of plasticity in deep continual learning. *Nat.* **2024**, *632*, 768–774.

(82) Wang, Y.; Fass, J.; Kaminow, B.; Herr, J. E.; Rufa, D.; Zhang, I.; Pulido, I.; Henry, M.; Chodera, J. D. End-to-End Differentiable Molecular Mechanics Force Field Construction. *Chem. Sci.* **2022**, *13*, 12016–12033.

(83) Rufa, D. A.; Macdonald, H. E. B.; Fass, J.; Wieder, M.; Grinaway, P. B.; Roitberg, A. E.; Isayev, O.; Chodera, J. D. Towards chemical accuracy for alchemical free energy calculations

with hybrid physics-based machine learning / molecular mechanics potentials. 2020; https://www.biorxiv.org/content/10.1101/2020.07.29.227959v1.

(84) Faber, E. B. et al. Screening through Lead Optimization of High Affinity, Allosteric Cyclin-Dependent Kinase 2 (CDK2) Inhibitors as Male Contraceptives That Reduce Sperm Counts in Mice. *J. of Med. Chem.* **2023**, *66*, 1928–1940.

(85) Fang, Y.; Zhang, Q.; Zhang, N.; Chen, Z.; Zhuang, X.; Shao, X.; Fan, X.; Chen, H. Knowledge graph-enhanced molecular contrastive learning with functional prompt. *Nat. Mach. Intell.* **2023**, *5*, 542–553.

(86) Hayes, T. et al. Simulating 500 million years of evolution with a language model. *Sci.* **2025**, *387*, 850–858.

TOC Graphic