PROFLOW: AN ITERATIVE REFINEMENT MODEL FOR PROTAC-INDUCED STRUCTURE PREDICTION

Bo Qiang
Department of Pharmaceutical Science
Peking University
Beijing, China
colinqiang@pku.edu.cn

Wenxian Shi
Department of Computer Science
Massachusetts Institute of Technology
Cambridge, MA, USA
wxsh@mit.edu

Yuxuan Song
Institute for Artificial Industry Research
Tsinghua University
Beijing, China
yxsong0816@gmail.com

Menghua Wu
Department of Computer Science
Massachusetts Institute of Technology
Cambridge, MA, USA
rmwu@mit.edu

ABSTRACT

Proteolysis targeting chimeras (PROTACs) are small molecules that trigger the breakdown of traditionally "undruggable" proteins by binding simultaneously to their targets and degradation-associated proteins. A key challenge in their rational design is understanding their structural basis of activity. Due to the lack of crystal structures (18 in the PDB), existing PROTAC docking methods have been forced to simplify the problem into a distance-constrained protein-protein docking task. To address the data issue, we develop a novel pseudo-data generation scheme that requires only binary protein-protein complexes. This new dataset enables PROFLOW, an iterative refinement model for PROTAC-induced structure prediction that models the full PROTAC flexibility during constrained protein-protein docking. PROFLOW outperforms the state-of-the-art across docking metrics and runtime. Its inference speed enables the large-scale screening of PROTAC designs, and computed properties of predicted structures achieve statistically significant correlations with published degradation activities.

1 INTRODUCTION

Targeted protein degradation is an emerging paradigm in rational drug design that induces the breakdown of "undruggable" proteins (Zhao et al., 2022). Proteolysis targeting chimeras (PROTACs) are small molecules that achieve this by simultaneously binding a protein of interest (POI) and a degradation-associated protein (e.g. E3 ligase) (Zou et al., 2019; Hu & Crews, 2022). In contrast to small molecule drugs, which attach to predefined sites on their protein targets, PROTACs operate by inducing a stable, *ternary* complex between themselves and two proteins which don't typically interact. This design task is highly structural, but modeling these structures has eluded both experimentalists and computationalists. There are only 18 such structures in the PDB (Weng et al., 2023), and PROTAC docking algorithms have yet to see widespread use (Troup et al., 2020), as existing search-based methods either are too slow or oversimplify the task. Finally, while deep learning has shown substantial promise in molecular docking (Corso et al., 2022), there are currently no end-to-end deep learning methods for PROTAC docking, due to the lack of data.

Specifically, given unbound 3D structures of the POI and E3 ligase and the 2D PROTAC molecular graph, our goal is to predict the bound poses of these three objects ("ternary complex," Figure 1). To this end, we propose PROFLOW, an iterative refinement model for PROTAC-induced structure prediction. We frame the task as a conditional generation problem, where we learn the distribution over rigid-body protein transformations that respect the existence of a connecting PROTAC linker. To train this model in the absence of real ternary complexes, we create a pseudo-ternary dataset using a novel data generation scheme that pairs binary protein-protein data with appropriate PROTAC linkers (Townshend et al., 2019; Weng et al., 2023).

Empirically, PROFLOW outperforms existing PROTAC docking methods in predicting PROTAC-induced complexes (8.35 interface RMSD) and E3-POI interfaces (0.264 Fnat). In addition, PROFLOW runs up to 60 times faster than the only alternative that considers full PROTAC structures, enabling the virtual screening of hundreds of designs within 5 hours. As a direct result, we are able to make predictions over the entire PROTAC-DB (Weng et al., 2021b) and show statistically-significant correlations between Rosetta-computed properties of our structures (Leaver-Fay et al., 2011) and published degradation activity. In summary, our main contributions are as follows.

1. To the best of our knowledge, PROFLOW is the first end-to-end deep learning approach for PROTAC-induced structure prediction. In contrast to previous work, we consider the full conformational landscape of the PROTAC linker during the entire sampling process.
2. We create and provide a new pseudo-ternary dataset from binary protein-protein complexes and PROTAC linker graphs, to facilitate further development of models for this task.
3. We achieve state-of-the-art performance on PROTAC docking benchmarks with up to 60 times speedup. In addition, computed properties of our predicted complexes show statistically significant correlations with published degradation activity, highlighting PROFLOW's utility for future design tasks.

2 Background and related work

Molecular docking Molecular docking plays a pivotal role in structure-based drug design (Morris & Lim-Wilby, 2008; Hwang et al., 2010). Given two or more molecules (e.g. small molecules and/or proteins), the goal is to predict their bound pose. Traditional docking algorithms break down the problem into two steps: a sampling method enumerates candidate 3D poses, and a scoring function ranks these poses (Trott & Olson, 2010; Yan et al., 2020). More recently, deep learning models have also been developed for molecular docking. These methods formulate the problem as either a regression (Stärk et al., 2022; Lu et al., 2022; Zhang et al., 2022) or a generative task (Corso et al., 2022; Ketata et al., 2023).

PROTAC docking Current methods for PROTAC docking incorporate PROTACs as constraints or filters for protein-protein docking. A common approach first samples PROTAC conformations and E3-POI complexes separately; then (optionally) reinserts and relaxes the PROTAC linkers to minimize collisions (Drummond & Williams, 2019; Drummond et al., 2020). This workflow has been used to query the structural bases of known PROTAC interactions (Weng et al., 2021a; Bai et al., 2021). The PROTAC can also be simplified into a set of distances for constrained protein-protein docking (Schneidman-Duhovny et al., 2005; Li et al., 2022). The primary drawback to these approaches is that they largely ignore the linker's flexibility by reducing it to a set of distances or fixed conformations during protein-protein sampling.

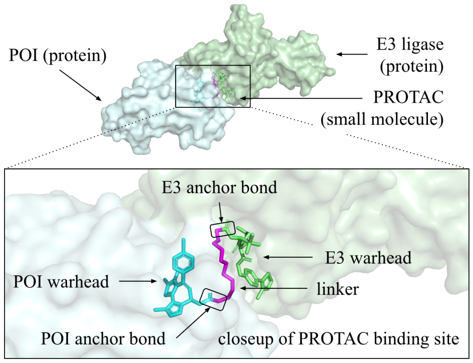
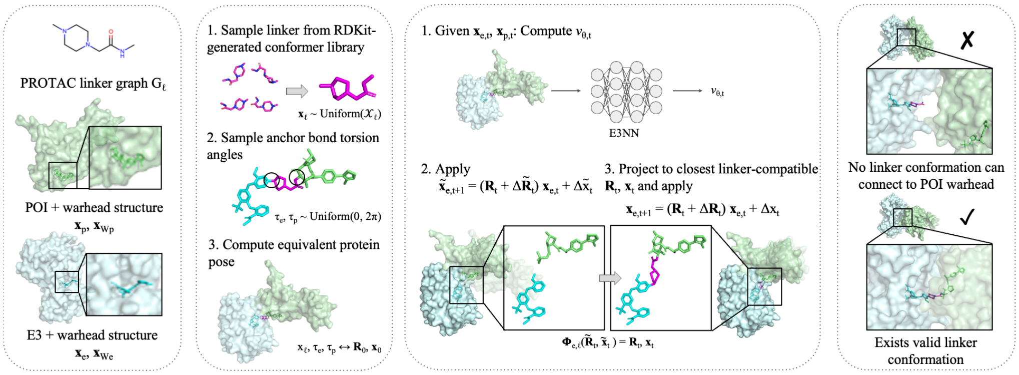


Figure 1: PROTACs are small molecules composed of two "warheads" and a connecting linker. The warheads bind to the E3 ligase and protein of interest (POI), while the flexible linker brings the two proteins into proximity. Top: ternary complex between POI (left), E3 ligase (right), PROTAC (small molecule). Bottom: PROTAC binding site, with respective warheads and anchor bonds. PDB 5T35.

Flow matching Flow matching (Lipman et al., 2023; Albergo et al., 2023) is an iterative refinement, generative modeling framework that has achieved the state-of-the-art across standard generation tasks (Tong et al., 2023) and biomolecular modeling (Stärk et al., 2023; Yim et al., 2023; Song et al., 2023). Flow matching has also been extended to Riemannian manifolds (Chen & Lipman, 2024). In this work, we develop an iterative refinement model based on flow matching, with chemically-informed modifications to ensure structural validity.


PROTAC linker graph G_ℓ
POI + warhead structure
$\mathbf{x}_p, \mathbf{x}_{Wp}$
E3 + warhead structure
$\mathbf{x}_e, \mathbf{x}_{We}$
1. Sample linker from RDKit-generated conformer library
$\mathbf{x}_\ell \sim \text{Uniform}(\mathcal{X}_\ell)$
2. Sample anchor bond torsion angles
$\tau_e, \tau_p \sim \text{Uniform}(0, 2\pi)$
3. Compute equivalent protein pose
$x_\ell, \tau_e, \tau_p \leftrightarrow \mathbf{R}_0, \mathbf{x}_0$
1. Given $\mathbf{x}_{e,t}, \mathbf{x}_{p,t}$: Compute $v_{\theta,t}$
E3NN
$v_{\theta,t}$
2. Apply
$\tilde{\mathbf{x}}_{e,t+1} = (\mathbf{R}_t + \Delta\tilde{\mathbf{R}}_t)\, \mathbf{x}_{e,t} + \Delta\tilde{x}_t$
3. Project to closest linker-compatible $\mathbf{R}_t$, $\mathbf{x}_t$ and apply
$\mathbf{x}_{e,t+1} = (\mathbf{R}_t + \Delta\mathbf{R}_t)\, \mathbf{x}_{e,t} + \Delta x_t$
$\mathbf{\Phi}_{e,\ell}(\tilde{\mathbf{R}}_t, \tilde{\mathbf{x}}_t) = \mathbf{R}_t, \mathbf{x}_t$
✗
No linker conformation can connect to POI warhead
✓
Exists valid linker conformation

(A) Model inputs

(B) Sample initial pose

(C) For t = 0 → 1: Integrate learned vector field and project to valid **R**, **x** between each step

(D) Linker-compatible space $\mathscr{M}_\ell$

Figure 2: Overview of PROFLOW, illustrated for PDB 7PI4. Iteratively refine E3 ligase poses by 1) predicting a rotation/translation update from the learned vector field, 2) applying these vectors to approximate the next pose, and 3) projecting to the closest linker-compatible pose.

The goal of flow matching is to learn a vector field u_t that transports a prior distribution $x_0 \sim q$ to the target data distribution $x_1 \sim p_{\text{data}}$. To efficiently learn u_t, we first define a *conditional flow map* $\psi_t(x_0 \mid x_1)$, whose derivative with respect to time yields the *conditional vector field* $\frac{d}{dt}\psi_t(x_0 \mid x_1) = u_t(x \mid x_1)$. The conditional vector field gives rise to a *conditional probability path* $p_t(x \mid x_1)$, $t \in [0, 1]$ via the continuity equation, where $p_0(x \mid x_1) = q(x)$ (prior) and $p_1(x \mid x_1) \approx \delta(x - x_1)$ (approximate Dirac).

The conditional flow matching loss is $\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{t,q(x),p_t(x|x_1)} \left\| v_\theta(x, t) - u_t\,(x \mid x_1) \right\|^2$ where v_θ is the output of a neural network θ. It has been shown that in expectation, regressing against conditional vector field $u_t(x \mid x_1)$ is equivalent to regressing against *marginal* vector field u_t (Tong et al., 2023). We can integrate Equation 2 over time to obtain p_{data} samples from noisy $x_0 \sim q$.

3 Methods

We present PROFLOW, an iterative refinement framework based on flow matching for PROTAC-induced structure prediction. As inputs, we are given unbound structures of an E3 ligase and protein of interest (POI), the molecular graph of the PROTAC, and (optionally) the E3 and POI warhead binding sites. Our goal is to sample the PROTAC-induced complex of the E3 ligase and POI (Figure 1).

Let $\mathbf{x}_e \in \mathbb{R}^{3 \times N_e}, \mathbf{x}_p \in \mathbb{R}^{3 \times N_p}$ denote the coordinates of the E3 ligase and POI. Let $G = (\mathcal{V}, \mathcal{E})$ denote the molecular graph of the PROTAC. G is associated with subgraphs G_{W_e}, G_{W_p}, G_ℓ, corresponding to the E3 warhead, POI warhead, and linker, where G_ℓ shares one edge ("anchor bond") with each of the warheads. We assume that the relative coordinates between warheads $\mathbf{x}_{W_e}, \mathbf{x}_{W_p}$ and the corresponding binding protein $\mathbf{x}_e, \mathbf{x}_p$ are given, either from ground truth or as the output of a protein-small molecule docking algorithm.[1] Our goal is to learn a generative model over $P(\mathbf{x}_e, \mathbf{x}_\ell \mid \mathbf{x}_p, G_\ell)$, where we arbitrarily fix the POI and transform the E3.

In practice, we approximate $P(\mathbf{x}_e, \mathbf{x}_\ell \mid \mathbf{x}_p, G_\ell)$ by sampling from $P(\mathbf{x}_e \mid \mathbf{x}_\ell, \mathbf{x}_p, G_\ell)$ and optimizing $P(\mathbf{x}_\ell \mid \mathbf{x}_e, \mathbf{x}_p, G_\ell)$. We learn the former using an iterative refinement model, introduced in the next section, and we compute the latter using a deterministic search algorithm over a large set of generated linker conformations (Appendix C.1).

3.1 PROFLOW framework

On a high level, our iterative refinement framework is as follows. For more details, see Appendix A.

1. We define a linker-compatible space $\mathcal{M}_\ell$, which is the space of E3 ligase transformations that can be "reached" by any conformation of the linker under consideration (Figure 2D).

[1] The former is a common assumption throughout the PROTAC docking literature (Zaidman et al., 2020), while the latter is empirically an easy task (Appendix D.3), as warheads are generally known binders.

Table 1: **Left:** re-docking, holo structures. **Right:** realistic docking, apo structures. **Format:** RMSD (%$<$ threshold, 20Å for cRMSD, 10Å for iRMSD). Methods marked * did not model the PROTAC. Runtimes marked † were performed exclusively on CPU (method does not support GPU). None of the methods utilize re-ranking or clustering for prediction selection.

	Holo structures						Apo structures				
Model	cRMSD (Å)		iRMSD (Å)		Fnat	Time(s)	cRMSD (Å)		iRMSD (Å)		Fnat
AF-Multimer*	14.83	(0.812)	10.49	(0.539)	0.221	735†	12.94	(0.865)	9.80	(0.563)	0.208
DiffDock-PP*	19.67	(0.510)	10.04	(0.509)	0.229	150	21.45	(0.380)	12.65	(0.446)	0.206
PROTACability	**12.45**	(0.926)	11.21	(0.496)	0.144	144†	27.58	(0.641)	25.67	0.275	0.118
PRossetaC	13.06	(0.954)	10.03	(0.487)	0.114	3015†	15.33	(0.856)	12.13	(0.367)	0.095
PROflow	12.85	**(0.975)**	**8.20**	**(0.740)**	**0.264**	**44**	**12.16**	**(0.995)**	**8.35**	**(0.643)**	**0.236**

2. We sample from our prior over $\mathcal{M}_\ell$ by sampling valid linker conformations $\mathbf{x}_\ell$, anchor bond torsion angles, and converting to equivalent protein transformations (Figure 2B). This ensures that all our samples remain within the linker-compatible space.
3. We define our conditional flow map using the objects in SE(3) flow matching (Yim et al., 2023), followed by a projection to $\mathcal{M}_\ell$ (Figure 2C).
4. Finally, we optimize an objective that compares the predicted transformation, to that yielded by the analytical conditional vector field in SE(3) flow matching.

Our learned vector field v_θ takes as input $G_\ell, \mathbf{x}_e, \mathbf{x}_p$, and time t. The model predicts as output $\hat{x}_{\theta,t}, \hat{R}_{\theta,t}$ in the tangent space to $\mathrm{SO}(3) \times \mathbb{R}^3$. We parametrize v_θ using E3NN, an equivariant graph convolution network (Geiger & Smidt, 2022). We process our inputs into protein interface point clouds, and include both geometric and chemical features. The final prediction aggregates all vertices into a virtual node and applies an additional tensor product filter.

3.2 Pseudo-data generation

So far, we have assumed that our model has access to triplets $(\mathbf{x}_e, \mathbf{x}_p, \mathbf{x}_\ell)$ for training. In reality, there are fewer than twenty ternary PROTAC structures in the PDB, so it is infeasible to train on existing data. Instead, we create a new pseudo-ternary dataset from binary protein-protein structures (DIPS) (Townshend et al., 2019) and known linker graphs (PROTAC-DB) (Weng et al., 2023).

First, for each of the 1507 unique linkers G_ℓ reported in Weng et al. (2023), we generate a set of 1024 conformations $\mathcal{X}_\ell$ using RDKit [2] (linker library). Next, for each binary protein-protein pair, we identify putative pocket candidates based on surface patches with high curvature. During training, we sample one pocket per protein, per pair, from the top 4 pockets with the highest curvature on each protein. Finally, for each pair of pockets, we identify an appropriate linker that has low RMSD to "anchor bonds," placed in the center of each pocket. See Appendix A.2 for more details. It's important to note that the training dataset and the ternary structures in our test set do not overlap.

4 Experiments

4.1 Docking experiments

Our test set consists of 13 solved PROTAC ternary structures from the PDB, downloaded from PROTAC-DB (Weng et al., 2023). We compare PROFLOW with three classes of docking algorithms.

- Unconditional protein-protein docking methods dock the E3 and POI without considering the PROTAC (Alphafold Multimer (Evans et al., 2021), DiffDock-PP (Ketata et al., 2023)).
- Warhead only: PROTACability (Pereira et al., 2023) assumes warhead binding sites are known for constrained protein-protein docking, but does not consider the linker.
- Full PROTAC: PRosettaC (Zaidman et al., 2020) assumes warhead binding sites are known and uses precomputed linker lengths as distance constraints for protein-protein docking.

[2] https://www.rdkit.org/

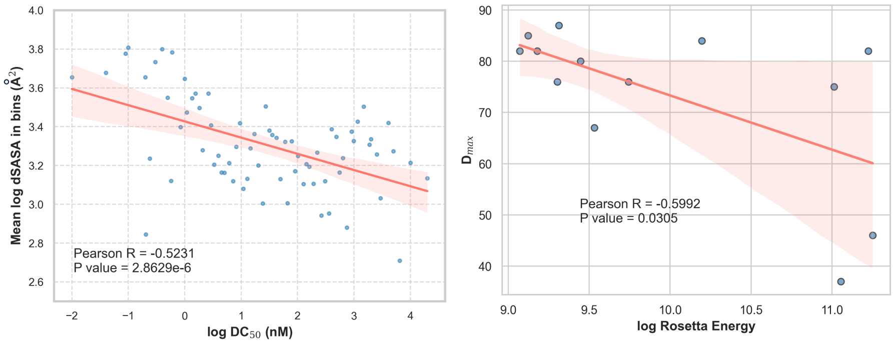


Figure 3: Left: DC$_{50}$ Solvent-accessible area buried at the interface(dSASA) of PROFLOW predictions, over PROTAC-DB. Right: D$_{\text{max}}$ Rosetta energy of PROFLOW predictions, over PROTACs that vary only in linker.

PROTACability and PRossetaC employ an additional clustering step tuned on part of our test set, so for fair comparison, we omit this in the main results. See Table 5 for results after post-processing.

Since PROTAC ternary structure prediction focuses primarily on docking the E3 and POI, we evaluate all methods based on common protein-protein docking metrics.

- Complex RMSD is determined by superimposing the predicted complex onto the ground truth complex using the Kabsch algorithm (Kabsch, 1976) and computing the RMSD between the α-carbons.
- Interface RMSD aligns only atoms in the ground truth binding interface (within 10Å of the binding partner) and computes the RMSD between their α-carbons.
- Fnat is the fraction of native interface contacts (within 15Å of the binding partner) preserved in the predicted complex's interface.

These thresholds are slightly higher than those in previous work (Ganea et al., 2021) since the distances between PROTAC complexes are longer than between natural protein binders (Figure 6).

Re-docking with holo structures We assess all models on their ability to recapitulate the E3-POI complex when provided their individual bound structures, as extracted from ternary structures (Table 1). PROFLOW outperforms all baselines in interface RMSD and Fnat, and achieves similar performance to the best baseline in complex RMSD. Of the baselines, those that do not consider the PROTAC at all (Alphafold-Multimer, DiffDock-PP) performed the worst in complex RMSD. We provide visual examples of our predictions in Figure 5. All models were given access to 10 CPU cores and 1 NVIDIA A6000 GPU for the runtime analysis. PROFLOW achieves the fastest runtime of all models, with an average of 44 seconds per complex.

Docking with apo structures Table 1 evaluates whether models are able to produce the POI-E3 complex, given only their unbound structures (apo-equivalents from the PDB) collected by Pereira et al. (2023). These sequences are nearly identical to their holo counterparts, with 95%+ sequence similarity. For evaluation, we aligned apo-equivalents to the holo structures and placed the warheads into the corresponding pockets. PROFLOW was able to maintain high performance, while all other baselines (that take structural inputs) experienced a notable drop. This result indicates their high reliance on holo conformations, whose availability is an unrealistic expectation, as very few E3s have been co-crystallized with non-native protein targets. The performance of AF-Multimer is improved mainly because of the alignment operation in apo structure docking procedure.

4.2 PROTAC DEGRADATION ACTIVITY EXPERIMENTS

A major goal of predicting PROTAC-induced ternary structures is to facilitate their rational design. Given that our model is able to predict high-quality structures at a fraction of the computational cost of all alternatives, while considering the entire PROTAC, we are able to predict and assess PROTAC structures at scale.

Large scale PROTAC-DB study We collected 483 examples with annotated DC_{50} values (the concentration of PROTAC required to degrade 50% of the POI) (Weng et al., 2023), whose unbound E3 and POI structures were available in the PDB. Nearly 70% of the PROTACs under consideration (334) share the same POI and E3 ligase with 4+ other PROTACs, so it is essential to model the linker in this analysis. Of our baselines, only PRosettaC (Zaidman et al., 2020) models the PROTAC linker. However, it was computationally intractable to benchmark PRosettaC at scale (nearly one hour per structure). We docked all warheads using Glide (Friesner et al., 2004). We sampled 20 PROFLOW structures per example and computed the solvent-accessible area buried at the interface (dSASA) using Rosetta (Leaver-Fay et al., 2011). We observed a statistically significant correlation between $\log$ dSASA and $\log \mathrm{DC}_{50}$ (p-value < 0.001) with a Pearson correlation coefficient of -0.5231 (Figure 3, left). The direction of this correlation is biologically intuitive: PROTACs that induce larger interfaces have lower DC_{50} and are more potent degraders.

VHL-SMARC2 case study In addition to DC_{50}, we also evaluate the relationship between the Rosetta energy of predicted PROTAC-induced E3-POI complexes and $\mathrm{D}_{\mathrm{max}}$ (maximum level of observed degradation). Data for 13 $\mathrm{D}_{\mathrm{max}}$ values were sourced from the study by Kofink et al. (2022). These PROTACs share identical warhead structures that target the same POI and E3 ligase and differ only in their linkers. Figure 3 (right) illustrates a statistically significant correlation (p-value < 0.05), Pearson coefficient -0.5992) between log Rosetta energy and $\mathrm{D}_{\mathrm{max}}$. Lower protein-protein energies signify a stronger POI and E3 ligase interaction, so PROFLOW predicted structures are able to filter out PROTAC designs with high degradation capacity.

5 CONCLUSION

In this work, we introduced PROFLOW, an iterative refinement model for PROTAC ternary structure docking. To train this model, we generated a new pseudo-ternary dataset from binary 3D protein-protein data and 2D linker graphs. Empirically, PROFLOW significantly outperformed baselines in interface RMSD and remained robust even when provided unbound structures. An order of magnitude faster than the best alternatives, PROFLOW enabled the screening of large PROTAC libraries, revealing correlations between computed properties and degradation activity. In future work, we hope to explore deep learning models for other modalities of targeted degradation, including molecular glues.

ACKNOWLEDGEMENTS

We thank Hannes Stark, Jason Yim, Bowen Jing, and Gabriele Corso for helpful discussions.

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. 1745302. We would like to acknowledge support from the NSF Expeditions grant (award 1918839: Collaborative Research: Understanding the World Through Code), Machine Learning for Pharmaceutical Discovery and Synthesis (MLPDS) consortium, the Abdul Latif Jameel Clinic for Machine Learning in Health, and the DTRA Discovery of Medical Countermeasures Against New and Emerging (DOMANE) threats program.

REFERENCES

Michael S. Albergo, Nicholas M. Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions, 2023.

Nan Bai, Sven A Miller, Grigorii V Andrianov, Max Yates, Palani Kirubakaran, and John Karanicolas. Rationalizing protac-mediated ternary complex formation using rosetta. *Journal of chemical information and modeling*, 61(3):1368–1382, 2021.

Ricky T. Q. Chen and Yaron Lipman. Flow matching on general geometries. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=g7ohDlTITL`.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. *arXiv preprint arXiv:2210.01776*, 2022.

Michael L Drummond and Christopher I Williams. In silico modeling of protac-mediated ternary complexes: validation and application. *Journal of chemical information and modeling*, 59(4): 1634–1644, 2019.

Michael L Drummond, Andrew Henry, Huifang Li, and Christopher I Williams. Improved accuracy for modeling protac-mediated ternary complex formation and targeted protein degradation via new in silico methodologies. *Journal of Chemical Information and Modeling*, 60(10):5234–5254, 2020.

Richard Evans, Michael O'Neill, Alexander Pritzel, Natasha Antropova, Andrew Senior, Tim Green, Augustin Žídek, Russ Bates, Sam Blackwell, Jason Yim, et al. Protein complex prediction with alphafold-multimer. *biorxiv*, pp. 2021–10, 2021.

Richard A Friesner, Jay L Banks, Robert B Murphy, Thomas A Halgren, Jasna J Klicic, Daniel T Mainz, Matthew P Repasky, Eric H Knoll, Mee Shelley, Jason K Perry, et al. Glide: a new approach for rapid, accurate docking and scoring. 1. method and assessment of docking accuracy. *Journal of medicinal chemistry*, 47(7):1739–1749, 2004.

Pablo Gainza, Freyr Sverrisson, Frederico Monti, Emanuele Rodola, D Boscaini, MM Bronstein, and BE Correia. Deciphering interaction fingerprints from protein molecular surfaces using geometric deep learning. *Nature Methods*, 17(2):184–192, 2020.

Octavian-Eugen Ganea, Xinyuan Huang, Charlotte Bunne, Yatao Bian, Regina Barzilay, Tommi Jaakkola, and Andreas Krause. Independent se (3)-equivariant models for end-to-end rigid protein docking. *arXiv preprint arXiv:2111.07786*, 2021.

Mario Geiger and Tess Smidt. e3nn: Euclidean neural networks. *arXiv preprint arXiv:2207.09453*, 2022.

Zhenyi Hu and Craig M Crews. Recent developments in protac-mediated protein degradation: from bench to clinic. *ChemBioChem*, 23(2):e202100270, 2022.

Howook Hwang, Thom Vreven, Joël Janin, and Zhiping Weng. Protein–protein docking benchmark version 4.0. *Proteins: Structure, Function, and Bioinformatics*, 78(15):3111–3114, 2010.

Bowen Jing, Bonnie Berger, and Tommi Jaakkola. Alphafold meets flow matching for generating protein ensembles. In *NeurIPS 2023 AI for Science Workshop*, 2023a.

Bowen Jing, Gabriele Corso, Jeffrey Chang, Regina Barzilay, and Tommi Jaakkola. Torsional diffusion for molecular conformer generation, 2023b.

Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923, 1976.

Mohamed Amine Ketata, Cedrik Laue, Ruslan Mammadov, Hannes Stärk, Menghua Wu, Gabriele Corso, Céline Marquet, Regina Barzilay, and Tommi S Jaakkola. Diffdock-pp: Rigid protein-protein docking with diffusion models. *arXiv preprint arXiv:2304.03889*, 2023.

Christiane Kofink, Nicole Trainor, Barbara Mair, Simon Wöhrle, Melanie Wurm, Nikolai Mischerikow, Michael J Roy, Gerd Bader, Peter Greb, Géraldine Garavel, et al. A selective and orally bioavailable vhl-recruiting protac achieves smarca2 degradation in vivo. *Nature Communications*, 13(1):5969, 2022.

Vincent Le Guilloux, Peter Schmidtke, and Pierre Tuffery. Fpocket: an open source platform for ligand pocket detection. *BMC bioinformatics*, 10(1):1–11, 2009.

Andrew Leaver-Fay, Michael Tyka, Steven M Lewis, Oliver F Lange, James Thompson, Ron Jacak, Kristian W Kaufman, P Douglas Renfrew, Colin A Smith, Will Sheffler, et al. Rosetta3: an object-oriented software suite for the simulation and design of macromolecules. In *Methods in enzymology*, volume 487, pp. 545–574. Elsevier, 2011.

Wenqing Li, Jiabin Zhang, Li Guo, and Qiantao Wang. Importance of three-body problems and protein–protein interactions in proteolysis-targeting chimera modeling: insights from molecular dynamics simulations. *Journal of Chemical Information and Modeling*, 62(3):523–532, 2022.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=PqvMRDCJT9t`.

Wei Lu, Qifeng Wu, Jixian Zhang, Jiahua Rao, Chengtao Li, and Shuangjia Zheng. Tankbind: Trigonometry-aware neural networks for drug-protein binding structure prediction. *Advances in neural information processing systems*, 35:7236–7249, 2022.

Garrett M Morris and Marguerita Lim-Wilby. Molecular docking. *Molecular modeling of proteins*, pp. 365–382, 2008.

Gilberto Pereira, Brian Jimenez Garcia, Riccardo Pellarin, Guillaume Launay, Sangwook Wu, Juliette Martin, and Paulo Cesar Telles de Souza. Rational prediction of protac-compatible protein-protein interfaces by molecular docking. *bioRxiv*, pp. 2023–02, 2023.

Michel F Sanner, Arthur J Olson, and Jean-Claude Spehner. Reduced surface: an efficient way to compute molecular surfaces. *Biopolymers*, 38(3):305–320, 1996.

Dina Schneidman-Duhovny, Yuval Inbar, Ruth Nussinov, and Haim J Wolfson. Patchdock and symmdock: servers for rigid and symmetric docking. *Nucleic acids research*, 33(suppl_2):W363–W367, 2005.

Yuxuan Song, Jingjing Gong, Minkai Xu, Ziyao Cao, Yanyan Lan, Stefano Ermon, Hao Zhou, and Wei-Ying Ma. Equivariant flow matching with hybrid probability transport for 3d molecule generation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Hannes Stärk, Octavian Ganea, Lagnajit Pattanaik, Regina Barzilay, and Tommi Jaakkola. Equibind: Geometric deep learning for drug binding structure prediction. In *International conference on machine learning*, pp. 20503–20521. PMLR, 2022.

Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Harmonic self-conditioned flow matching for multi-ligand docking and binding site design, 2023.

Freyr Sverrisson, Jean Feydy, Bruno E Correia, and Michael M Bronstein. Fast end-to-end learning on protein surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15272–15281, 2021.

Alexander Tong, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Kilian Fatras, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport, 2023.

Raphael Townshend, Rishi Bedi, Patricia Suriana, and Ron Dror. End-to-end learning on 3d protein structure for interface prediction. *Advances in Neural Information Processing Systems*, 32, 2019.

Oleg Trott and Arthur J Olson. Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of computational chemistry*, 31(2):455–461, 2010.

Robert I Troup, Charlene Fallan, and Matthias GJ Baud. Current strategies for the design of protac linkers: a critical review. *Exploration of Targeted Anti-tumor Therapy*, 1(5):273, 2020.

Gaoqi Weng, Dan Li, Yu Kang, and Tingjun Hou. Integrative modeling of protac-mediated ternary complexes. *Journal of Medicinal Chemistry*, 64(21):16271–16281, 2021a.

Gaoqi Weng, Chao Shen, Dongsheng Cao, Junbo Gao, Xiaowu Dong, Qiaojun He, Bo Yang, Dan Li, Jian Wu, and Tingjun Hou. Protac-db: an online database of protacs. *Nucleic acids research*, 49 (D1):D1381–D1387, 2021b.

Gaoqi Weng, Xuanyan Cai, Dongsheng Cao, Hongyan Du, Chao Shen, Yafeng Deng, Qiaojun He, Bo Yang, Dan Li, and Tingjun Hou. Protac-db 2.0: an updated database of protacs. *Nucleic acids research*, 51(D1):D1367–D1372, 2023.

Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. Geodiff: a geometric diffusion model for molecular conformation generation, 2022.

Yumeng Yan, Huanyu Tao, Jiahua He, and Sheng-You Huang. The hdock server for integrated protein–protein docking. *Nature protocols*, 15(5):1829–1852, 2020.

Jason Yim, Andrew Campbell, Andrew YK Foong, Michael Gastegger, José Jiménez-Luna, Sarah Lewis, Victor Garcia Satorras, Bastiaan S Veeling, Regina Barzilay, Tommi Jaakkola, et al. Fast protein backbone generation with se (3) flow matching. *arXiv preprint arXiv:2310.05297*, 2023.

Daniel Zaidman, Jaime Prilusky, and Nir London. Prosettac: Rosetta based modeling of protac mediated ternary complexes. *Journal of chemical information and modeling*, 60(10):4894–4903, 2020.

Yangtian Zhang, Huiyu Cai, Chence Shi, Bozitao Zhong, and Jian Tang. E3bind: An end-to-end equivariant network for protein-ligand docking. *arXiv preprint arXiv:2210.06069*, 2022.

Lin Zhao, Jia Zhao, Kunhong Zhong, Aiping Tong, and Da Jia. Targeted protein degradation: mechanisms, strategies and application. *Signal Transduction and Targeted Therapy*, 7(1):113, 2022.

Yutian Zou, Danhui Ma, and Yinyin Wang. The protac technology in drug development. *Cell biochemistry and function*, 37(1):21–30, 2019.

A ITERATIVE REFINEMENT FRAMEWORK

PROTAC-linker compatible space It is non-trivial to include G_ℓ explicitly in the model. However, generating multiple conformations $\mathcal{X}_\ell$ based on G_ℓ is an easy task (Jing et al., 2023b). In addition, the linker primarily functions as a positional scaffold and is less involved in the physicochemical interactions specific to the elements. Therefore, we approach G_ℓ conditional modeling alternatively by defining the space of SE(3) transformations of the E3 ligase in the following space that respect a set of chemically plausible conformations $\mathcal{X}_\ell$ of the linker (Figure 2D).

Definition A.1 (Linker-compatible space)**.** Let $\mathbf{x}_{[\cdot]-\ell}$ denote the structure of a protein-warhead complex connected to linker ℓ at its respective anchor bond, and let $\mathbf{x}_{[\cdot]}[b] \in \mathbb{R}^{3\times 2}$ denote the coordinates that correspond to anchor bond b.

For a given $\mathbf{x}_e, \mathbf{x}_p$ and G_ℓ, a transformation $(R, x) \in$ SE(3) belongs to *linker-compatible space* $\mathcal{M}_\ell$ if there exists a conformation $\mathbf{x}_\ell$ that connects the POI and E3 under (R, x):

$$\exists \mathbf{x}_\ell \in \mathcal{X}_\ell,\ \ (R\,\mathbf{x}_{e-\ell} + x)\,[b_p] = \mathbf{x}_p[b_p], \tag{1}$$

where b_p is the POI anchor bond.

It is non-trivial to define the standard conditional flow matching objects within $\mathcal{M}_\ell$. Instead of operating directly over this set, we leverage the objects of SE(3) flow matching Yim et al. (2023) and project back into $\mathcal{M}_\ell$ at each step — a procedure we call *pseudo*-flow matching.

Similar to standard flow matching, we first define a prior distribution on $\mathcal{M}_\ell$. Then, we define the *pseudo*-conditional flow map from the prior to the target data distribution by interpolating on SE(3) and projecting the intermediate points from SE(3) to $\mathcal{M}_\ell$. Finally, we discuss our modified training objective and touch upon symmetry considerations.

Prior distribution To design a flow matching framework, we require an easy-to-sample $q_\ell(z)$ over $\mathcal{M}_\ell$. However, it is not necessarily easy to sample from this space: selecting an arbitrary $(R, x) \in$ SE(3) does not ensure the existence of a compatible $\mathbf{x}_\ell$.

Therefore, we take an inverted approach: instead of directly operating over the protein, we first sample a valid linker conformation, and then compute the equivalent protein transformation (Figure 2B). To summarize, sampling from q_ℓ can be decomposed into three steps.

1. Sample linker conformation $\mathbf{x}_\ell \sim$ Uniform$(\mathcal{X}_\ell)$.
2. Sample torsion angles for each anchor bond $\tau_e, \tau_p \sim$ Uniform$(0, 2\pi)$.
3. Map to equivalent E3 ligase transformation (R, x).

The closed-form solution for the mapping $\mathbf{x}_\ell, \tau_e, \tau_p \mapsto (R, x)$ can be obtained by absorbing the anchor bond torsion angles (τ_e, τ_p) and angles within the linker's local structure into rotation matrices. A detailed derivation is presented in Section B.1.

Intuitively, this procedure can be interpreted as sampling from the linker conformation energy-weighted distribution over $\mathcal{M}_\ell$. Transformations (R, x) that correspond to stable linker conformations are sampled more frequently, and (R, x) that result from different torsion angle settings of the same linker conformation are sampled equally often.

Pseudo-conditional flow map Interpolating along the geodesic paths in SO(3) and $\mathbb{R}^3$ gives rise to the following conditional flow maps Yim et al. (2023):

$$\tilde{R_t} = \exp_{R_0}(t \log_{R_0}(R_1)) \tag{2}$$

$$\tilde{x_t} = (1-t)x_0 + t \cdot x_1. \tag{3}$$

However, this interpolation does not guarantee that we remain within $\mathcal{M}_\ell$. Instead, we introduce the following projection to define a reasonable path within $\mathcal{M}_\ell$.

Definition A.2 (Projection to $\mathcal{M}_\ell$)**.** Given $(R, x) \in$ SE(3), we define $\Phi_{\ell,e}$: SE$(3) \to \mathcal{M}_\ell$

$$\Phi_{\ell,e}(R, x) = \underset{(R^*, x^*)\in\mathcal{M}_l}{\arg\min}\ \text{RMSD}\left((R\mathbf{x}_e + x), (R^*\mathbf{x}_e + x^*)\right) \tag{4}$$

where RMSD$(\mathbf{x}, \mathbf{x}') = \sqrt{1/n \cdot \sum_i^n \|\mathbf{x}_i - \mathbf{x}'_i\|^2}$.

This procedure allows us to compare different transformations based on their effects on $\mathbf{x}_e$, and to implicitly update the $\mathbf{x}_\ell$ and τ_e, τ_p under consideration at each step. We implement this mapping via the conformer matching optimization procedure used in Jing et al. (2023b) (Appendix C.1).

We define the pseudo-conditional flow map over $\mathcal{M}_\ell$,

$$(R_t, x_t) = \Phi_{\ell,e}(\tilde{R}_t, \tilde{x_t}) \tag{5}$$

where $\tilde{R}_t, \tilde{x_t}$ are given by Equations (2, 3). At $t = 0$, our linker-centric sampling procedure ensures $(R_0, x_0) \in \mathcal{M}_\ell$, and at $t = 1$, (R_1, x_1) are computed directly from data.

Training objective Due to the nature of the projection to $\mathcal{M}_\ell$, it is difficult to obtain an analytical formula for the vector field u_t. If we first ignore the projection, standard SE(3) flow matching yields a closed form solution by differentiating Equations (2, 3) with respect to t Yim et al. (2023):

$$\dot{x}_t \approx \frac{x_1 - \tilde{x}_t}{1 - t}, \qquad \dot{R}_t \approx \frac{\log_{\tilde{R}_t}(R_1)}{1 - t}. \tag{6}$$

The simplest solution is to apply the standard conditional flow matching loss (Equation 2) to these terms. However, this approach leads to unstable performance, due to the irregularity of $\mathcal{M}_\ell$ (analysis in Appendix D.4.3).

Therefore, we optimized an alternative, which compares the effect of predicted transformations on the coordinates. Let $z \sim q$ and $(R_t, x_t) \sim p_t(x \mid z)$, then:

$$\tilde{\mathbf{x}}_{e,t+\Delta t} = \dot{R}_t \Delta t (R_t \mathbf{x}_{e,t}) + (x_t + \dot{x}_t \Delta t) \tag{7}$$

$$\hat{\mathbf{x}}_{e,t+\Delta t} = \hat{R}_{\theta,t} \Delta t (R_t \mathbf{x}_{e,t}) + (x_t + \hat{x}_{\theta,t} \Delta t) \tag{8}$$

where $\hat{x}_{\theta,t}, \hat{R}_{\theta,t}$ are the outputs of the learned vector field v_θ, and Δt is set to a small constant. We minimize

$$\mathcal{L}_{\mathcal{M}_l} = \mathbb{E}_{t,q(z),p_t(x|z)} \text{RMSD}(\tilde{\mathbf{x}}_{e,t+\mathrm{d}t}, \hat{\mathbf{x}}_{e,t+\mathrm{d}t}). \tag{9}$$

Sampling During inference, we integrate (2) from $t \in [0, 1]$ using an Euler solver (Chen et al., 2018). We apply $\Phi_{\ell,e}$ between each step to ensure that our transformations remain within $\mathcal{M}_\ell$ (Figure 2C and Algorithm 1).

SE(3) equivariance To ensure that our likelihood is SE(3)-equivariant, we require a SE(3) invariant prior and SE(3)-equivariant updates Xu et al. (2022). Our prior distribution is SE(3) invariant since the torsion angles of anchor bonds τ_e, τ_p are SE(3) invariant, we express $\mathbf{x}_\ell$ relative to the input $\mathbf{x}_p$ with local structures fixed. The learned vector field v_θ is parametrized by a SE(3) equivariant architecture based on Geiger & Smidt (2022).

A.1 Model implementation

Input representations Given the two input protein structures, we first compute their surfaces using MSMS and triangulate them into point clouds of vertices (same as in Section 3.2). We featurize each vertex with geometric and chemical features (Gainza et al., 2020; Sverrisson et al., 2021), including its coordinates, curvature, normal vector, 6 closest non-hydrogen atoms, and the vectors between these atoms and the vertex itself.

Prior work has shown that when protein-protein interfaces are known, removing non-interface residues from consideration increases both computational efficiency and docking performance (Pereira et al., 2023). Since warhead poses are determined ahead of time, we follow the same approach and discard any vertices more than 20Å from the surface of the binding partner, at initialization. Though this definition of interface depends on the initial $\mathbf{x}_\ell$, this threshold preserves approximately one-third of most proteins, which is usually a superset of the actual interface.

SE(3)-equivariant architecture We implement v_θ using a 6-layered E3NN, a SE(3)-equivariant graph convolutional network (Geiger & Smidt, 2022). Within each protein point cloud, we draw edges between the 3 nearest neighbors of each vertex, and across the protein interface, we add edges

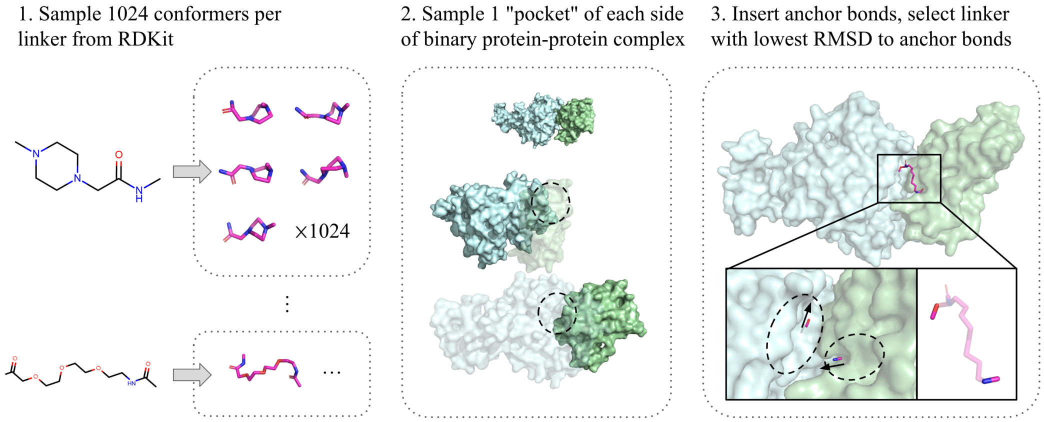

1. Sample 1024 conformers per linker from RDKit
2. Sample 1 "pocket" of each side of binary protein-protein complex
3. Insert anchor bonds, select linker with lowest RMSD to anchor bonds
×1024

Figure 4: Pseudo-data generation procedure. 1) Generate PROTAC linker conformation library. 2) Identify high-curvature putative pockets and sample 1 per protein in binary protein-protein complex. 3) Match each protein pair to linker with lowest RMSD from anchor bonds.

between vertices within 15Å apart. To produce the SE(3)-equivariant outputs $\hat{x}_{\theta,t}, \hat{R}_{\theta,t}$ from the final layer, we aggregate information from all vertices into a virtual node, placed at the center of all E3 ligase α-carbon atoms (including those outside the interface), and apply an additional tensor product filter. Additional information can be found in Appendix C.3.

Steric guidance Though our E3NN focuses exclusively on the protein interface surface, we would like to avoid steric clashes from parts of the proteins outside of their interfaces. We add an additional potential to the gradient to bias the model away from transformations that would otherwise cause clashes. We modify our vector field as follows,

$$u_t(x) \leftarrow u_t(x) + \sigma_t \nabla_x \sum_{i=1}^{m_p} \sum_{j=1}^{m_e} [-\max(0, d_{ij} - \rho)], \tag{10}$$

where $\sigma_t = \sigma_{\min}^{1-t} \cdot \sigma_{\max}^{t}$ is an exponential scheduler, d_{ij} is the Euclidean distance between α-carbons of residues i, j, and $\rho = 2$Å is the contract threshold.

A.2 Data generation procedure

Linker conformation library For each of the 1507 unique linkers G_ℓ reported in Weng et al. (2023), we generate a set of 1024 conformations $\mathcal{X}_\ell$ using RDKit [3].

Pseudo-binding sites We identify putative warhead binding sites on each side of the binary protein-protein interface. Since protein interfaces are too flat for standard pocket prediction methods Le Guilloux et al. (2009), we opt for a simpler geometric approach. We compute each protein's surface using MSMS Sanner et al. (1996) and triangulate with a vertex density of 0.5/Å^2. Putative pockets are obtained by clustering the top 200 vertices with highest curvature, based on pairwise Euclidean distance. Pseudo-anchor bonds are inserted at each cluster's center of mass, pointing towards the average of the vertices' normal vectors. For each protein, we extract the top 4 highest curvature clusters as pockets. During training, we sample one pocket per protein uniformly at random, from a total of 16 pairs.

Linker matching For each pair of protein pockets, we select the best-matched linker from our conformation library $\{\mathcal{X}_\ell\}$ based on Kabsch alignment of the anchor bonds Kabsch (1976). While this step only considers the two endpoints of each linker, most PROTAC linkers tend to be simple chains, which rarely result in any clashes with the protein complexes.

In total, this process allowed us to create approximately 685k triplets (42.8k protein pairs in DIPS × 16 pocket combinations per protein pair). Our model was trained on these pseudo-data from DIPS and finetuned on a similarly created dataset of binary E3 ligase structures (Appendix C.2).

[3] https://www.rdkit.org/

B Mathematical Derivations

B.1 Solution for equivalent prior distribution sampling

In Section A, we decompose the sampling from prior distribution into sampling linker conformation and torsion angles. Here, we introduced how to derive the equivalent transformation (R, x) from linker conformation $\mathbf{x}_\ell$, τ_e and τ_p. At first, we reduce the unnecessary degrees of freedom in $\mathbf{x}_\ell$ by representing it as the m torsion angles within the linker rotatable bonds b_r as $\tau_\ell \in SO(2)^m$.

Because linkers usually adopt linear molecular graphs, the rotation matrices could be used for updating the ternary structure in sequential order. We reorder the atoms in the linker according to the distance to the POI side anchor bond to adopt the following permutation:

$$\forall i, j : i < j \Rightarrow D_{\mathcal{G}_\ell}(i, 0) < D_{\mathcal{G}_\ell}(j, 0) \tag{11}$$

$D_{\mathcal{G}}(i, j)$ is defined as the shortest path length between node i and node j in graph $\mathcal{G}$. The anchor bond from the POI side is denoted as edge $(0, 1)$ in the $\mathcal{G}_\ell$. In addition, we also reordered the rotatable bonds as:

$$b_{r_i} = (i_1, i_2), \tag{12}$$

$$b_{r_j} = (j_1, j_2) \tag{13}$$

$$\forall i, j : i < j \Rightarrow (i_1 < j_1 \text{ and } i_2 < j_2) \tag{14}$$

Then, we transform all angles into rotation matrices:

$$R(\mathbf{n}, \alpha) = \begin{bmatrix} \mathbf{n}_x^2 + (1 - \mathbf{n}_x^2)\cos\alpha & \mathbf{n}_x\mathbf{n}_y(1 - \cos\alpha) - \mathbf{n}_z\sin\alpha & \mathbf{n}_x\mathbf{n}_z(1 - \cos\alpha) + \mathbf{n}_y\sin\alpha \\ \mathbf{n}_x\mathbf{n}_y(1 - \cos\alpha) + \mathbf{n}_z\sin\alpha & \mathbf{n}_y^2 + (1 - \mathbf{n}_y^2)\cos\alpha & \mathbf{n}_y\mathbf{n}_z(1 - \cos\alpha) - \mathbf{n}_x\sin\alpha \\ \mathbf{n}_x\mathbf{n}_z(1 - \cos\alpha) - \mathbf{n}_y\sin\alpha & \mathbf{n}_y\mathbf{n}_z(1 - \cos\alpha) + \mathbf{n}_x\sin\alpha & \mathbf{n}_z^2 + (1 - \mathbf{n}_z^2)\cos\alpha \end{bmatrix} \tag{15}$$

$$R_{\ell_i} = R(\frac{\mathbf{x}_\ell[b_{r_i}]}{\|\mathbf{x}_\ell[b_{r_i}]\|}, \tau_{\ell_i}), R_{\tau_e} = R(\frac{\mathbf{x}_{W_e}[b]}{\|\mathbf{x}_{W_e}[b]\|}, \tau_e), R_{\tau_p} = R(\frac{-\mathbf{x}_{W_p}[b]}{\|\mathbf{x}_{W_p}[b]\|}, \tau_e) \tag{16}$$

The transformation of rotating a substructure $\mathbf{x}[\cdot]$ around the rotation axis $\mathbf{x}[i, j]$ with angle τ can be written as:

$$Trans(\tau, \mathbf{x}[i, j])\mathbf{x}[\cdot] = R(\frac{\mathbf{x}[i, j]}{\|\mathbf{x}[i, j]\|}, \tau)(\mathbf{x}[\cdot] - \mathbf{x}[i]) + R(\frac{\mathbf{x}[i, j]}{\|\mathbf{x}[i, j]\|}, \tau)\mathbf{x}[i] \tag{17}$$

We denote given initial structure as $\mathbf{x}_{e_{int}}$. As a result, mapping the sampled linker conformation $\mathbf{x}_\ell$ and torsion angles (τ_e, τ_p) into the E3 ligase relative position $\mathbf{X}_e$ formalizes a closed function as:

$$\mathbf{x}_e = [Trans(\tau_e, \mathbf{x}_{W_e}[b]) \cdot Trans(\tau_{\ell_m}, \mathbf{x}_\ell[b_{r_m}]) \cdots Trans(\tau_{\ell_0}, \mathbf{x}_\ell[b_{r_0}]) \cdot Trans(\tau_p, \mathbf{x}_{W_p}[b])]\mathbf{x}_{e_{int}} \tag{18}$$

C Implementation Details

C.1 Search algorithm for linker matching

In this section, we introduce the algorithm applied for $P(\mathbf{x}_\ell \mid \mathbf{x}_e, \mathbf{x}_p, G_\ell)$ and $\Phi_{\ell,e}$. Though these 2 processes are utilized in different steps of the generation, a conditional conformation generation can be implemented to solve both. According to the definition of anchor bonds, their coordinates $\mathbf{x}_\ell[b]$ are determined when $\mathbf{x}_W[b]$ is determined by $\mathbf{x}_e$ and $\mathbf{x}_p$ because we keep the warhead-protein complex structure fixed.

Both $P(\mathbf{x}_\ell \mid \mathbf{x}_e, \mathbf{x}_p, G_\ell)$ and $\Phi_{\ell,e}$ can be reduced to searching process for finding $\mathbf{x}_\ell \in \mathcal{X}_\ell$ to minimize the RMSD objective in Equation 4. The optimum for this function is guaranteed to be 0 for $P(\mathbf{x}_\ell \mid \mathbf{x}_e, \mathbf{x}_p, G_\ell)$, because $\mathbf{x}_e$ lies in the constraint subspace, $\mathcal{M}_\ell$. However, for $\Phi_{\ell,e}$, the optimum is not guaranteed to be 0. As a result, the position of E3 ligase is updated in $\Phi_{\ell,e}$ to ensure $\text{RMSD}(\mathbf{x}_\ell[b], \mathbf{x}_W[b]) = 0$ by aligning $\mathbf{x}_{W_e}[b]$ with $\mathbf{x}_\ell[b_e]$.

Therefore, to minimize the objective, $\text{RMSD}\left((R\mathbf{x}_e + x), (R^*\mathbf{x}_e + x^*)\right)$, s.t. $(R^*, x^*) \in \mathcal{M}_\ell$ we apply a scoring function with a optimization algorithm. To score a given linker structure $\mathbf{x}_\ell^*$, we first align the $\mathbf{x}_\ell^*[b]$ with $\mathbf{x}_W[b]$ derived from $\mathbf{x}_e$ and $\mathbf{x}_p$. We add interpolation coordinates within $\mathbf{x}_\ell^*[b_p]$ and $\mathbf{x}_W[b_p]$ to strengthen the weighting in the Kabsch alignment, as we want to achieve precise alignment of $\mathbf{x}_\ell^*[b_p]$ and $\mathbf{x}_W[b_p]$. This step is pivotal for evaluation on $\mathbf{x}_\ell^*[b_e]$ and $\mathbf{x}_W[b_e]$ from the E3 ligase side, since our RMSD is defined for E3 ligase coordinates $\mathbf{x}_e$. We denote the function of calculating the centroid coordinate as c, then R^*, x^* is written as:

$$R^* = R(\frac{\overrightarrow{\mathbf{x}_\ell^*[b_e]} \times \overrightarrow{\mathbf{x}_W[b_e]}}{\|\overrightarrow{\mathbf{x}_\ell^*[b_e]}\| \, \|\overrightarrow{\mathbf{x}_W[b_e]}\|}, \arccos(\frac{\overrightarrow{\mathbf{x}_\ell^*[b_e]} \cdot \overrightarrow{\mathbf{x}_W[b_e]}}{\|\overrightarrow{\mathbf{x}_\ell^*[b_e]}\| \, \|\overrightarrow{\mathbf{x}_W[b_e]}\|})) \tag{19}$$

$$x^* = R^*(c(\mathbf{x}_e) - c(\mathbf{x}_W[b_e])) - (c(\mathbf{x}_e) - c(\mathbf{x}_W[b_e])) + (c(\mathbf{x}_\ell^*[b_e]) - c(\mathbf{x}_W[b_e])) \tag{20}$$

At each scoring step, we calculate the above (R^*, x^*) to the RMSD evaluation in Equation 4. We represent $\mathbf{x}_\ell$ as the torsion angles τ_ℓ and optimize the angles using an optimization algorithm given the above scoring function. To balance between accuracy and efficiency, the DIRECT algorithm is applied for Φ_l when we solve the ODE for sampling, and the evolution algorithm is applied for $P(\mathbf{x}_\ell \mid \mathbf{x}_e, \mathbf{x}_p, G_\ell)$.

C.2 Procedure for protein-protein training dataset curation

In this section, we introduce the procedure for preparing the binary protein-protein data for the data generation pipeline (Section 3.2). We mainly train our data on the Database of Interacting Protein Structures (DIPS) (Townshend et al., 2019) dataset and finetune it on the E3 ligase dataset.

For the DIPS dataset, we simply run the script from the official repository [4]. Unlike protein-protein docking researches that assign the larger protein chain as receptor and the shorter as ligand, we randomly assign (receptor, ligand) to the chains to increase the size of training data and because of the fact that E3 ligase is not always smaller than the POI. Though there are already 42,826 data points within the DIPS dataset, we still want to curate a dataset that only focuses on protein-protein complexes of E3 ligase + X, where X could be any protein chain. Firstly, we retrieve all the protein structures with multiple chains from PDB by using the E3 ligase Uniprot ID from PROTAC-DB (Weng et al., 2021b). Then we screened these PDB files by extracting the E3 ligase chain and keeping the surrounding protein chain if it possesses more than 40 residues and more than 5 residues are close to the E3 ligase chain by the threshold of 8Å. By extracting the protein chain files, we curated a protein-protein dataset of 883 E3-ligase+X data. In the training, the E3 ligase is treated as the ligand and the binding chain is assigned as the pseudo-POI.

C.3 SE(3) model architecture details

As outlined in Section A, our parametrized vector field is implemented using a SE(3) convolutional network, following the architecture of the score model in Diffdock (Corso et al., 2022). In the following sections, we denote $\otimes_W$ as the spherical tensor product with weights W and $\oplus$ as the normal vector addition. The architecture is decomposed into three main layers: the embedding layer, interaction layer, and output layer, as detailed below.

Embedding layer Though previous protein-protein docking works (Ganea et al., 2021; Ketata et al., 2023) prefer to use residue-types and the coordinates of α-carbons to represent rigid protein structures, PROTAC docking requires fine-grained representations as the protein-protein interface are predetermined by the warhead positions. In our work, we represent protein as surface representations. At first, we process the chemical features of the nearest atom into the atom embedding and vertex representations (curvature, size, type, normal vector) calculated by MSMS into the vertex embeddings with the same size of atom embedding. Both atom embedding and vertex embedding include scalar representations and vector representations. Then we aggregate 6 atom embeddings into the corresponding vertex embeddings by the following fully connected tensor product:

$$\mathbf{h}_v = \frac{1}{6+1} \sum_{\mathbf{h}_i \in (\mathbf{h}_a, \mathbf{h}_v)} \sum_{\mathbf{h}_j \in (\mathbf{h}_a, \mathbf{h}_v)} \mathbf{h}_i \otimes_W \mathbf{h}_j \tag{21}$$

In the embedding layer, W are trainable parameters. In order to make PROFLOW scalable to large protein-protein interfaces, we then cluster the vertices into a fixed number(256 in our model) of

[4] https://github.com/drorlab/DIPS

patches. The patch representations are calculated by averaging the vertex representations and place in the centroid of the corresponding vertices.

Interaction layer We defined the mechanism for updating the patch representations following the convolutional network in Diffdock (Corso et al., 2022). For a given patch with representation $\mathbf{h}_p$ that is connected to $\mathcal{N}_p$ other patches, for each message passing procedure, the representation is updated as:

$$\mathbf{h}_p \leftarrow \mathbf{h}_p \oplus \mathbf{BN}\left(\frac{1}{|\mathcal{N}_a|}\sum_{n\in\mathcal{N}_a} Y\left(r_{np}\right) \otimes_{\psi_{np}} \mathbf{h}_n\right), \psi_{np} = \Psi(\mathbf{h}_p, \mathbf{h}_n, \|r_{np}\|^2) \tag{22}$$

Here, Y represents spherical harmonics, and BN stands for equivariant batch normalization. Ψ denotes the trainable MLP responsible for computing the weights for the tensor product. In practice, we construct a k-nearest-neighbor graph (k=3) for message passing within patches for the same protein and a radius graph with a cutoff of 15Å for message passing between patches belonging to the POI and E3 ligase, respectively. In each interaction layer, the representations are initially updated by aggregating information from the same protein and subsequently exchanging information across proteins.

Output layer It was crucial for us to aggregate the information back to the coordinate system of the full protein from representations from protein surfaces because the training objective is defined on full protein α-carbon coordinates. An additional virtual node is placed at the centroid of E3 ligase α-carbons. For all the patches of E3 ligase $\mathcal{V}_e$ and the centroid virtual node v_c, the message is passed to v_c as:

$$\mathbf{v_c} \leftarrow \frac{1}{|\mathcal{V}_e|}\sum_{n\in\mathcal{V}_e} Y\left(\hat{r}_{cn}\right) \otimes_{\psi_{cn}} \mathbf{h}_n, \psi_{cn} = \Psi(\mathbf{h}_n, \|r_{cn}\|^2) \tag{23}$$

To parametrize the vector field in the tangent space of SE(3), $\mathbf{v_c}$ is a single odd and a single even vector. We sum the odd representation with the even representation as the final prediction. For loss computation in training, we update the E3 ligase coordinates with our prediction and the ground truth score. The RMSD between the score updated coordinate and the predicted score updated coordinate is our training objective.

Algorithm 1 Sampling algorithm of PROFLOW

```
Input: Initial $\mathbf{x}_p$ and $G_\ell$
Output: Bound $\mathbf{x}_e, \mathbf{x}_\ell$
Sample from prior distribution $p_0$: $\mathbf{x}_\ell \sim \text{Uniform}(\mathcal{X}_\ell)$, $\tau_e, \tau_p \sim \text{Uniform}(0, 2\pi)$
Map to equivalent $x_0 = (R_0, X_0)$
for t in 0, $\Delta t, 2\Delta t, \cdots 1$ do
    Compute $\hat{R}_t, \hat{X}_t$ from learned vector field $v_\theta$
    Update and refine to $\mathcal{M}_\ell$: $X_t \rightarrow \Phi_{\ell,e}(\hat{R}_t \Delta t x_t + \hat{X}_t \Delta t)$
    Compute the gradient of rotation and translation of $u(t)(X_t)$ in Eq.10: $\hat{R}_{st}$,$\hat{X}_{st}$
    Update and refine to $\mathcal{M}_\ell$: $X_t \rightarrow \Phi_{\ell,e}(\hat{R}_{st} \Delta t x_t + \hat{X}_{st} \Delta t)$
end for
$\mathbf{x}_e = x_1$ and search $\mathbf{x}_\ell$ using the method described in Sec. C.1
```

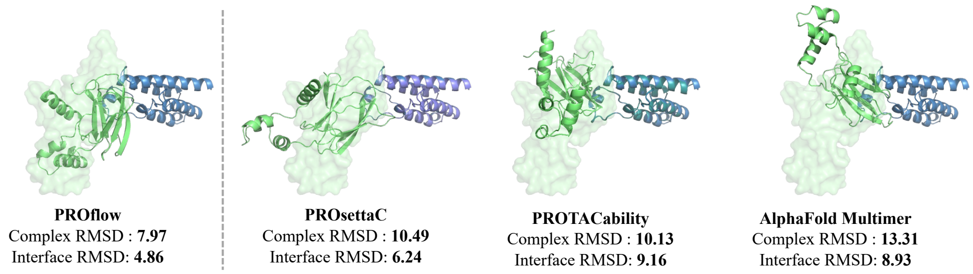
D Additional Experimental Results

D.1 Visual examples

D.2 Protein-protein interface distance distribution

D.3 Warhead docking analysis

Experimental setup To demonstrate the capability of existing search-based software in accurately predicting the relative structures of protein-warhead for PROTAC ternary structure docking, we curated a dataset from PROTAC-DB (Weng et al., 2021b) and PDB for warhead docking experiments.



PROflow
Complex RMSD : **7.97**
Interface RMSD: **4.86**

PROsettaC
Complex RMSD : **10.49**
Interface RMSD: **6.24**

PROTACability
Complex RMSD : **10.13**
Interface RMSD: **9.16**

AlphaFold Multimer
Complex RMSD : **13.31**
Interface RMSD: **8.93**

Figure 5: Visualized results of structures (PDB 6HAX) predicted by different methods. The ground truth E3 ligase pose is represented as a semitransparent surfaces.

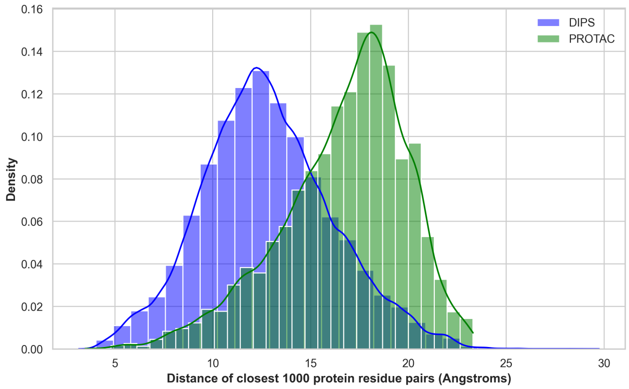


Figure 6: Distribution of the residue distances for the 1,000 closest pairs, drawn from the protein-protein binary dataset (DIPS) and the PROTAC test set.

Initially, we retrieved E3 ligase and POI structures from PDB using the UniProt ID of existing PROTACs from PROTAC-DB. Subsequently, we filtered out all monomer proteins lacking binding small-molecule ligands. Among the remaining small molecules binding to POI or E3-ligase, we retained data where the maximum common structure with PROTAC warheads accounted for more than 80% of the ligands. Finally, we removed all solvent molecules from the structures, keeping only the chains binding with the ligands. 2245 pairs of proteins and warheads-like small molecules are collected after this procedure.

Because of this large amount of data retrieved from PDB, we could draw a conclusion that the pocket pockets on existing POIs and E3 ligases are fully explored. Even if certain targets are considered undruggable by traditional inhibitors, it is not inherently challenging to locate a binder within the PDB to establish a preliminary understanding of the binding pocket's approximate location. Therefore, we mainly focused on pocket-specific docking instead of blind docking. We employed the Schrodinger suite for evaluating docking on our curated dataset and 13 ternary PROTAC structures. We initially prepared each protein structure by preprocessing them with the Protein Preparation Wizard. This step involved refining the protein structure, adding hydrogen atoms, and assigning bond orders. Subsequently, the ligand structures were processed through LigPrep to generate appropriate ionization and tautomeric states. The receptor grid of size 20Å×20Å×20Å was generated around the active site using the Receptor Grid Generation tool. For docking simulations, we employed Glide, Schrödinger's molecular docking program, applying standard precision (SP) and extra precision (XP) protocols. The SP protocol was employed for initial virtual screening, followed by more accurate XP docking to refine the binding poses and rank the ligands.

Result and discussion As presented in Table 2, the existing method demonstrates proficiency in predicting over half of the warhead-like structures within 2Å, with a median RMSD of 1.59Å after the molecular docking. This substantiates the accurate prediction of protein-warhead structures for a broad spectrum of candidate warhead structures suitable for PROTAC design. Subsequently, we evaluate the docking accuracy of the protein-warhead structures in our test set. The median RMSD for docking warheads is 0.962Å. Additionally, since PROFLOW relies on the positions of anchor

Table 2: The results of RMSD and centroid distance in warhead-pocket docking experiments. We present the docking accuracy obtained from two datasets: the curated PROTAC-DB dataset and 13 PROTAC ternary structures sourced from PDB. The metrics are computed based on full warhead structures or anchor bonds.

Dataset	Test component	Size of test set	Mean(Å) ↓	Median(Å) ↓	<2Å ↑
PROTAC-DB	full warhead (RMSD)	2245	2.86	1.59	0.565
ternary complexes	full warhead (RMSD)	26	2.51	0.962	0.714
	anchor bond (RMSD)	26	1.94	0.774	0.750
	anchor bond centroids	26	0.769	0.27	0.893

Table 3: Experimental results for PROTAC ternary structure docking using ground truth structures of protein-warhead and Glide-docked structures of protein-warhead.

	Complex RMSD(Å)			Interface RMSD(Å)			Fnat	
Methods	Mean ↓	Median ↓	<20 ↑	Mean ↓	Median ↓	<10 ↑	Mean ↓	Median ↓
Docked warheads	13.01	13.51	0.985	8.51	8.52	0.658	0.231	0.194
True warhead-protein	12.85	13.12	0.975	8.20	8.16	0.740	0.264	0.211

bonds to maintain linker structures, we assess the docking accuracy of anchor bonds. The median centroid distances between the predicted anchor bonds and ground truth anchor bonds are 0.27Å, indicating a precise prediction of anchor bond positions. Finally, we present the experimental results of PROFLOW using docked warheads instead of ground truth in Table D.3. Despite a slight decline in performance, the decrease in metrics is relatively small when compared to the variations observed when employing different methods.

D.4 Ablation study

D.4.1 Ablation on training dataset and sampling

Table 4: Ablation study results presenting the impact of removing part of training data and steric guidance in sampling in the implementation of PROFLOW.

	Complex RMSD(Å)			Interface RMSD(Å)			Fnat	
Methods	Mean ↓	Median ↓	<20 ↑	Mean ↓	Median ↓	<10 ↑	Mean ↓	Median ↓
DIPS only	13.10	13.24	0.981	7.90	7.82	0.690	0.301	0.262
E3 ligase dataset only	13.12	13.18	0.992	8.48	8.60	0.750	0.231	0.194
No steric guidance	14.33	14.67	0.911	8.81	9.12	0.688	0.241	0.189
Full PROFLOW	12.85	13.12	0.975	8.20	8.16	0.740	0.264	0.211

In this ablation study, we explored the impact of removing a portion of the training data and incorporating the steric guidance for sampling. Notably, when we only train PROFLOW on DIPS or E3 ligase dataset, the model's performance in complex RMSD exhibited a reduction. Regarding the interface RMSD and Fnat, the model achieved better results by only using the DIPS dataset. Therefore, we can conclude that the general dataset, DIPS provides more information on protein-protein interaction on the surface level. In order to balance the performance variation on complex RMSD and interface RMSD, we choose to use both training data sets for our main experiment. Additionally, the incorporation of the steric guidance technique demonstrated effectiveness in boosting the performance from the full complex level. These findings underscore the importance of training data and sampling strategies.

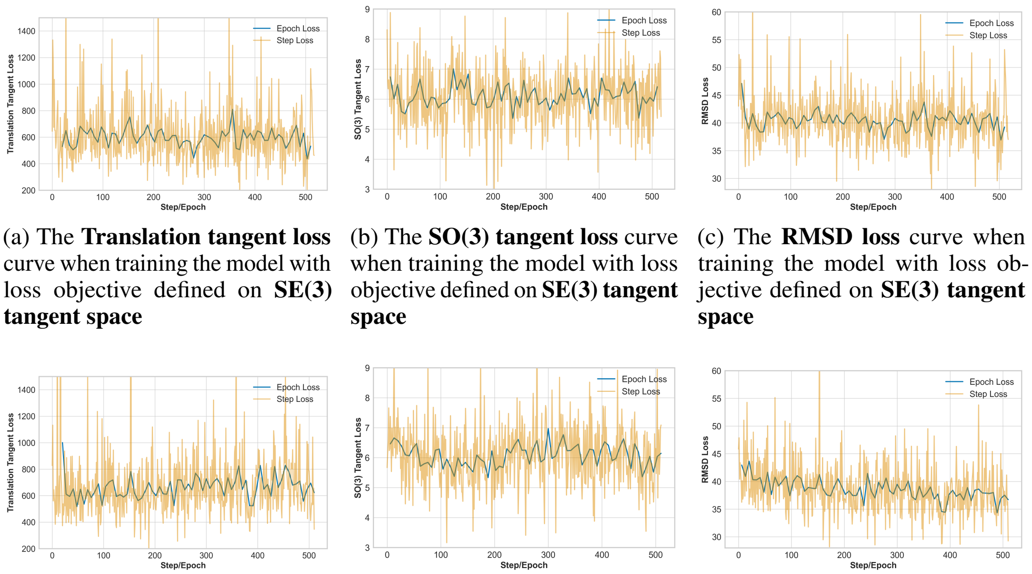
D.4.2 Baseline results with ranking and clustering

In our main experimental results, we mainly focused on the docking accuracy of the sampled structures of PROflow and baseline methods, so we excluded the postprocessing procedure of PROsettac. Here we downloaded the top-cluster structures and top-scored structures from the official PROsettac repository [5] after the ranking and clustering. They experimented on 7 unique structures downloaded from the PDB. We provided the results in Table 5.

Table 5: Ablation study on comparing PROflow with PRosettaC with clustering and ranking

	Complex RMSD(Å)			Interface RMSD(Å)			Fnat
Methods	Mean ↓	Med ↑	<20 ↑	Mean ↓	Med ↑	<10 ↑	Mean ↓
PRosettaC(top1 cluster)	10.38	10.39	1.000	8.19	8.21	0.543	0.284
PRosettaC(top20% Rosetta score)	10.23	10.30	0.996	7.79	7.60	0.610	0.283
PROflow(no ranking/clustering)	12.55	12.60	0.963	8.01	8.08	0.829	0.313

D.4.3 Stability of using different training objective



(a) The **Translation tangent loss** curve when training the model with loss objective defined on **SE(3) tangent space**

(b) The **SO(3) tangent loss** curve when training the model with loss objective defined on **SE(3) tangent space**

(c) The **RMSD loss** curve when training the model with loss objective defined on **SE(3) tangent space**

(d) The **Translation tangent loss** curve when training the model with **RMSD objective on the Euclidean space**

(e) The **SO(3) tangent loss** curve when training the model with **RMSD objective on the Euclidean space**

(f) The **RMSD loss** curve when training the model with **RMSD objective on the Euclidean space**

Figure 7: The loss curve of PROflow by using different training objective

In this section, we analyze the loss landscape (Figure 7) corresponding to different training objectives. Figures 7(a), (b), and (c) depict the training loss curves for employing the training objective within the tangent space of SO(3) × $\mathbb{R}^3$, as previous work by Yim et al. (2023). The translation loss in $\mathbb{R}^3$ and rotation loss in SO(3) exhibit noise, making it challenging to monitor their convergence. Figures 7(d), (e), and (f) depict the training loss curves for employing our RMSD loss. Conversely, the loss curve in Figure 7(f) demonstrates more stable convergence in comparison to the conventional equivariant flow matching loss for rotation and translation. Simultaneously, when utilizing the RMSD loss, the translation and rotation losses continue to converge within the noisy landscape.

[5] https://github.com/LondonLab/PRosettaC

From a theoretical point of view, the training objective could be defined as the distance measure in three different spaces including Euclidean space $\mathbb{R}^{3\times N}$, the SE(3) manifold, and linker constraint space $\mathcal{M}_\ell$. They are hierarchically related as subspaces, which means that $\mathcal{M}_\ell \subset$ SE(3) $\subset \mathbb{R}^{3\times N}$. The RMSD loss we use corresponds to the distance in Euclidean space, where optimizing it is equivalent to optimizing a weighted sum of the distances in $\mathbb{R}^3$ and SO(3). The weight is automatically adjusted by the inner distances of E3 ligase coordinates.

Traditional CFM loss treats SE(3) as the product space of $\mathbb{R}^3$ and SO(3) where the training objective is separated. From the above loss analysis, dividing SE(3) into translation and rotation is hard for the optimizer. Our training objective draws inspiration from prior work (Jing et al., 2023a), which transitions the flow matching loss from the space $\mathbb{R}^{3\times N}$ to the quotient space $\mathbb{R}^{3\times N}/SE(3)$. In PROFLOW, the loss is lifted into $\mathbb{R}^{3n}$ to jointly optimize the loss objective on translation space and rotation space that demonstrates practical efficacy.