DESIGN OF LIGAND-BINDING PROTEINS WITH ATOMIC FLOW MATCHING

Junqi Liu[1,3], **Shaoning Li**[2,3], **Chence Shi**[3,4,5], **Zhi Yang**[1,†], **Jian Tang**[4,6,7,3,†]
[1] School of Computer Science, Peking University [2] The Chinese University of Hong Kong
[3] BioGeometry [4] Mila - Québec AI Institute [5] Université de Montréal [6] HEC Montréal
[7] CIFAR AI Research Chair
{liujunqi,yangzhi}@pku.edu.cn
jian.tang@hec.ca

ABSTRACT

Designing novel proteins that bind to small molecules is a long-standing challenge in computational biology, with applications in developing catalysts, biosensors, and more. Current computational methods rely on the assumption that the binding pose of the target molecule is known, which is not always feasible, as conformations of novel targets are often unknown and tend to change upon binding. In this work, we formulate proteins and molecules as unified biotokens, and present ATOMFLOW, a novel deep generative model under the flow-matching framework for the design of ligand-binding proteins from the 2D target molecular graph alone. Operating on representative atoms of biotokens, ATOMFLOW captures the flexibility of ligands and generates ligand conformations and protein backbone structures iteratively. We consider the multi-scale nature of biotokens and demonstrate that ATOMFLOW can be effectively trained on a subset of structures from the Protein Data Bank, by matching flow vector field using an SE(3) equivariant structure prediction network. Experimental results show that our method can generate high-fidelity ligand-binding proteins and achieve performance comparable to the state-of-the-art model RFDiffusionAA, while not requiring bound ligand structures. As a general framework, ATOMFLOW holds the potential to be applied to various biomolecule generation tasks in the future.

1 INTRODUCTION

Proteins are essential macromolecules that drive vital biological processes, primarily through binding with small molecules (Schreier et al., 2009). While progress has been made in designing ligand-binding proteins for functions like catalysis and biosensing (Bennett et al., 2023), challenges persist due to complex protein-molecule interactions and ligand flexibility. Traditional approaches rely on shape complementarity to dock molecules onto native protein scaffolds (Bick et al., 2017; Polizzi & DeGrado, 2020), but these methods are computationally expensive.

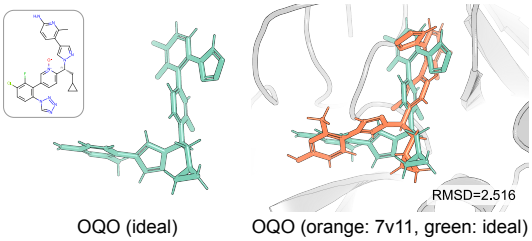

Figure 1: The conformer of OQO deforms upon binding to coagulation factor XIa. Green: ideal conformer. Orange: bound conformer.

Recently, RFDiffusionAA (Krishna et al., 2024), built on the all-atom structure prediction model RoseTTAFoldAA (Krishna et al., 2024), has demonstrated superior performance in designing ligand-binding proteins by explicitly modeling protein-molecule interactions. However, current methods assume a known, rigid binding conformation, which is often unavailable for molecules lacking natural protein-binding partners (Bick et al., 2017). Sampling diverse conformers and filtering them with expert knowledge (Krishna et al., 2024) is computationally demanding, while enforcing ligand rigidity is suboptimal, as ligands typically undergo conformational changes upon binding (Mobley & Dill, 2009). Figure.1 illustrates this phenomenon. Although recent

methods (Zhang et al., 2024; Stark et al., 2024) address ligand flexibility, they only design protein regions that directly interact with ligands and require the rest of the protein as input.

To address the aforementioned issues, we present Atomic Flow-matching (ATOMFLOW), a novel deep generative model grounded in the flow-matching framework (Lipman et al., 2022) for the design of ligand-binding proteins from 2D molecular graphs alone. We follow the *in silico* evaluation pipeline of the state-of-the-art method RFDiffusionAA, evaluating ATOMFLOW on several key metrics including self-consistency, binding affinity, diversity, and novelty. ATOMFLOW matches the overall performance of RFDiffusionAA and demonstrates advantages in various situations. An ablation study further highlights that when the bound structure is unknown, ATOMFLOW successfully designs protein binders with high binding affinity, whereas RFDiffusionAA can be constrained by its dependence on a fixed, suboptimal ligand structure.

2 RELATED WORK

Ligand-binding Protein Design. Traditional ligand-binding protein design relies on docking molecules into shape-complementary protein pockets (Polizzi & DeGrado, 2020; Lu et al., 2024). While deep learning accelerates screening (An et al., 2023), conventional methods remain computationally expensive and expert-dependent (Bick et al., 2017). Recent deep generative models offer data-driven alternatives, designing proteins conditioned on binding targets (Shi et al., 2022; Kong et al., 2023; Watson et al., 2023; Zhang et al., 2024). For molecule binder design, RFDiffusion (Watson et al., 2023) generates proteins using a heuristic potential for shape complementarity, while RFDiffusionAA (Krishna et al., 2024) improves performance by explicitly modeling protein-molecule interactions. However, these methods assume fixed ligand binding poses and impose rigidity constraints. Other approaches focus on designing binding pockets while considering ligand flexibility (Stark et al., 2024; Zhang et al., 2024), but they only modify protein regions interacting with ligands. Our model, by contrast, fully designs ligand-binding proteins from 2D molecular graphs while accounting for ligand flexibility.

Protein Generative Model and Structure Prediction. Recent deep generative models advance protein design (Ingraham et al., 2023; Lin & AlQuraishi, 2023; Yim et al., 2023b;a; Wu et al., 2024; Watson et al., 2023; Krishna et al., 2024). For example, Genie (Lin & AlQuraishi, 2023) diffuses Cα coordinates, while FrameDiff (Yim et al., 2023b) extends this to SE(3) diffusion on residue frames. FrameFlow (Yim et al., 2023a) accelerates this process via flow matching. However, these models focus on single-chain proteins and lack multi-molecule modeling. Our approach treats proteins and molecules as biotokens, applying flow-matching on representative atoms to design ligand-binding proteins from molecular graphs, capturing biomolecular flexibility and interactions. Related structure prediction methods like RoseTTAFoldAA (Krishna et al., 2024) and AlphaFold 3 (Abramson et al., 2024) tokenize biomolecules for universal modeling. Our ATOMFLOW follows this practice while emphasizing biointeraction patterns through structural modeling on key atoms, enhancing protein-molecule information flow (Bryant et al., 2024).

3 PRELIMINARIES

Notations. We represent a protein-ligand complex as a series of N biotokens $\{a_i \mid a_i = (s_i, x_i), i = 1, 2, \ldots, N\}$, where a_i corresponds to either a protein residue or ligand atom. s_i denotes the token type, and $x_i \in \mathbb{R}^3$ is the token position. Let $\mathcal{S}_{\text{protein}}$ and $\mathcal{S}_{\text{atom}}$ be the amino acid types and chemical elements, respectively. For protein residues, $s_i \in \mathcal{S}_{\text{protein}}$, and x_i is the C-α position; for ligand atoms, $s_i \in \mathcal{S}_{\text{atom}}$, and x_i is the atomic position. We define the protein token set as $\mathcal{P}$ and the ligand token set as $\mathcal{M}$, where $N_p = |\mathcal{P}|$ and $N_m = |\mathcal{M}|$, with $N = N_p + N_m$. Each biotoken has token-level features $f^{\text{token}} \in \mathbb{R}^{N \times c_t}$ and pair-level features $f^{\text{pair}} \in \mathbb{R}^{N \times N \times c_p}$, where c_t and c_p are feature dimensions.

Problem Formulation. Given a ligand molecule as a chemical graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a protein residue count N_p, we aim to generate a protein-ligand complex. The goal is to generate the token positions $\{x_i\}$, where $\mathbf{x}_m = \{x_i \mid a_i \in \mathcal{M}\}$ represents a valid conformer of $\mathcal{G}$, and $\mathbf{x}_p = \{x_i \mid a_i \in \mathcal{P}\}$ represents a protein binder with high binding affinity. Following prior work (Krishna et al., 2024; Yim et al., 2023b), we also generate protein token frames $\{T_i = (r_i, t_i) \mid a_i \in \mathcal{P}\}$, which recover full backbone coordinates. Residue type design $\{s_i \mid a_i \in \mathcal{P}\}$ is handled by a reverse folding model (Dauparas et al., 2023).

4 METHOD

ATOMFLOW uses a unified *biotoken* representation to jointly generate protein and ligand structures by learning the distribution of token positions conditioned on a ligand chemical graph $\mathcal{G}$. Figure 2 illustrates the overall framework. We introduce a rectified flow on token positions $\mathbf{x} \in \mathbb{R}^{N\times 3}$ and approximate its vector field with an SE(3)-equivariant structure prediction network. Below, we summarize key points of our method.

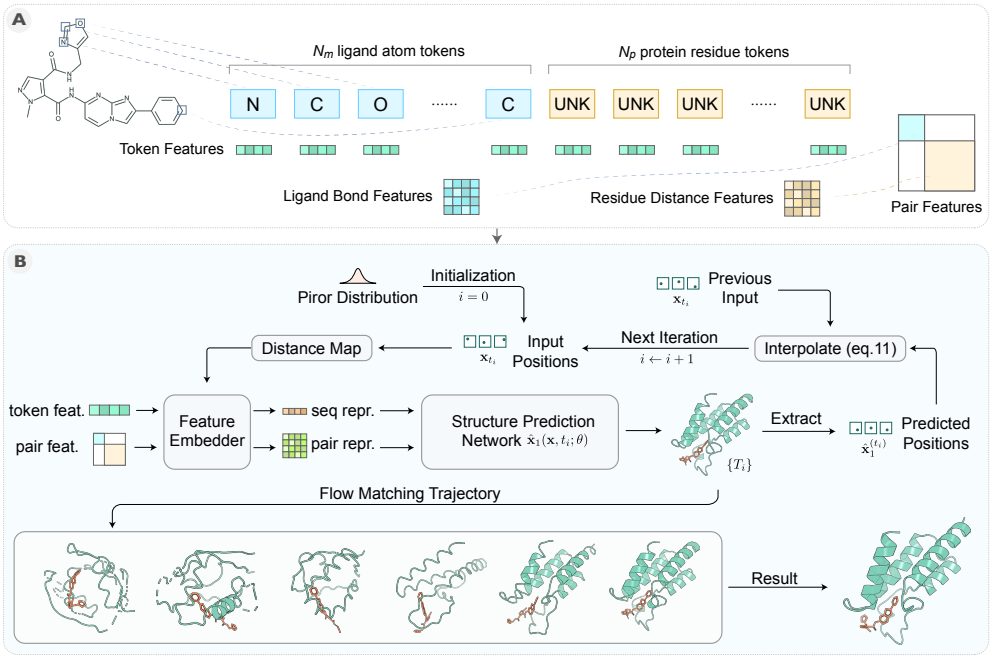

Figure 2: Overview of ATOMFLOW. Biotokens (protein residues and ligand atoms) are jointly embedded. A flow matching procedure gradually refines noisy coordinates into a plausible complex structure, using an SE(3)-equivariant network.

4.1 FLOW MATCHING FOR PROTEIN-LIGAND COMPLEX GENERATION

We jointly generate the protein-ligand complex structure $\mathbf{x} = \mathbf{x}_m \cup \mathbf{x}_p$ in $\mathbb{R}^{N\times 3}$, treating any two structures that can be aligned by an SE(3) transformation as identical elements in the quotient space $\mathcal{Q}$. To make training tractable, we define a rectified flow on $\mathcal{Q}$ using a conditional vector field:

$$u_t(\mathbf{x} \mid \mathbf{x}_1) = \frac{1}{1-t}\Big(\text{align}_{\mathbf{x}}(\mathbf{x}_1) - \mathbf{x}\Big), \tag{1}$$

where $\mathbf{x}_1$ is the target structure from the data distribution, and $\text{align}_{\mathbf{x}}(\mathbf{x}_1)$ is its best RMSD alignment to $\mathbf{x}$. We train the network to approximate $u_t(\mathbf{x} \mid \mathbf{x}_1)$ by minimizing:

$$\mathcal{L}_{\text{CFM}-\text{FAPE}}(\theta) = \mathbb{E}_{t, p_{\text{data}}(\mathbf{x}_1), p_t(\mathbf{x}|\mathbf{x}_1)}\Big[\frac{1}{1-t}\text{FAPE}\big(\hat{\mathbf{x}}_1(\mathbf{x}, t; \theta), \mathbf{x}_1\big)\Big]. \tag{2}$$

Here, $\hat{\mathbf{x}}_1$ is predicted by our structure network. With approximation (Appendix A.5), we find it more numerically stable to use an FAPE-based loss, with does not change the final training target (Jumper et al., 2021): We partition the FAPE loss into protein-protein, protein-ligand, and ligand-ligand interactions with appropriate scaling factors.

4.2 BIOTOKEN REPRESENTATION AND CONDITIONING

We embed both protein residues and ligand atoms into a unified token space. For ligand tokens, we encode atomic elements and known chemical properties in f^{token}, while bond connectivity forms f^{pair}. Protein residues incorporate positional indices and any additional known constraints into these

features. This unified representation lets the transformer-based network handle protein and ligand tokens jointly, without separate processing pipelines.

4.3 Structure Prediction Network

The SE(3)-equivariant network $\hat{\mathbf{x}}_1(\mathbf{x}, t; \theta)$ takes noisy coordinates $\mathbf{x}$, time t, and token/pair embeddings, then predicts a set of local frames (one per token). We ultimately extract each token's 3D position from its predicted frame. We encode input coordinates via a binned distance map, which is SE(3)-invariant and accommodates both atomic and residue-scale distances. We transform token- and pair-level features (including t) into single and pair representations (s, z), concatenating relevant inputs before a series of linear projections. A transformer stack refines the pair representation z, which then feeds into an invariant point attention (IPA) module (Jumper et al., 2021) to output token frames. For ligand atoms, the rotation is identity (only translation is used). The final structure is thus an SE(3)-aware prediction of all token coordinates. Additionally, we add a distance prediction head from the refined pair representation. The resulting cross-entropy loss on binned distances provides an additional training signal that helps stabilize learning.

4.4 Training and Inference

We draw training samples $(\mathbf{x}_1, \mathcal{G})$ and randomly sample $t \in [0, 1]$. We mix $\mathbf{x}_1$ with a noise sample from the prior distribution $q(\mathbf{x})$ to obtain a noisy input. The network is trained by minimizing the FAPE-based conditional flow matching loss in Equation 2. For inference, we start with a random sample $\mathbf{x}_0 \sim q(\mathbf{x})$ and numerically integrate the learned vector field from $t = 0$ to $t = 1$ using an ODE solver (e.g., Euler's method). At each step, we predict $\hat{\mathbf{x}}_1$, align it, and update $\mathbf{x}$ accordingly. The final structure at $t = 1$ is our designed protein-ligand complex.

5 Experiments

Following prior work on protein design (Yim et al., 2023a; Lin & AlQuraishi, 2023; Watson et al., 2023) and binder design (Krishna et al., 2024), we evaluate our method (AtomFlow) through *in silico* experiments on key binder metrics: self-consistency, binding affinity, diversity, and novelty.

5.1 Experiment Setup

Training Data. We train our denoising model on two datasets: PDBBind (Liu et al., 2017), a protein-ligand conformer set from the Protein Data Bank (PDB) (Berman et al., 2000), and SCOPe (Chandonia et al., 2022), a protein structure classification set. The model is first trained on protein-only generation for 400k steps, then finetuned to co-generate proteins and ligands for 300k steps.

Baselines and Variants. We compare AtomFlow to RFDiffusionAA (Krishna et al., 2024), which requires a *fixed* ligand structure as input. We train two AtomFlow variants: AtomFlow-N (no structure hint) and AtomFlow-H (with pairwise distance hints for ligand atoms). We exclude PocketGen (Zhang et al., 2024) and FlowSite (Stark et al., 2024) because they only refine existing pockets; see Appendix for additional experiments.

Evaluation Set. Following (Krishna et al., 2024), we evaluate on ligands FAD, SAM, IAI, and OQO (some seen during training, some not). Each ligand also varies in length. We further test on an extended set in Appendix A.6.

5.2 Self-consistency and Conformer Legitimacy

We assess output validity by (1) *self-consistency RMSD (scRMSD)* for protein structures and (2) detecting any structural violation in the generated ligand conformer. Self-consistency measures how well a generated structure agrees with the fold predicted by an external model. We follow (Lin & AlQuraishi, 2023; Watson et al., 2023), generating sequences via LigandMPNN (Dauparas et al., 2023), folding them with ESMFold (Lin et al., 2023), and comparing the folded structure to the model output. We sample 10 structures for each ligand in {100,150,200,250,300} residues.Figure 3A,D show scRMSD results. Additional plots are in Appendix A.6.

AtomFlow (AtomFlow-N, AtomFlow-H) and RFDiffusionAA outperform RFDiffusion, indicating that direct protein-ligand interaction modeling improves structure quality. AtomFlow-H slightly benefits from the distance hint, but AtomFlow-N remains highly competitive.

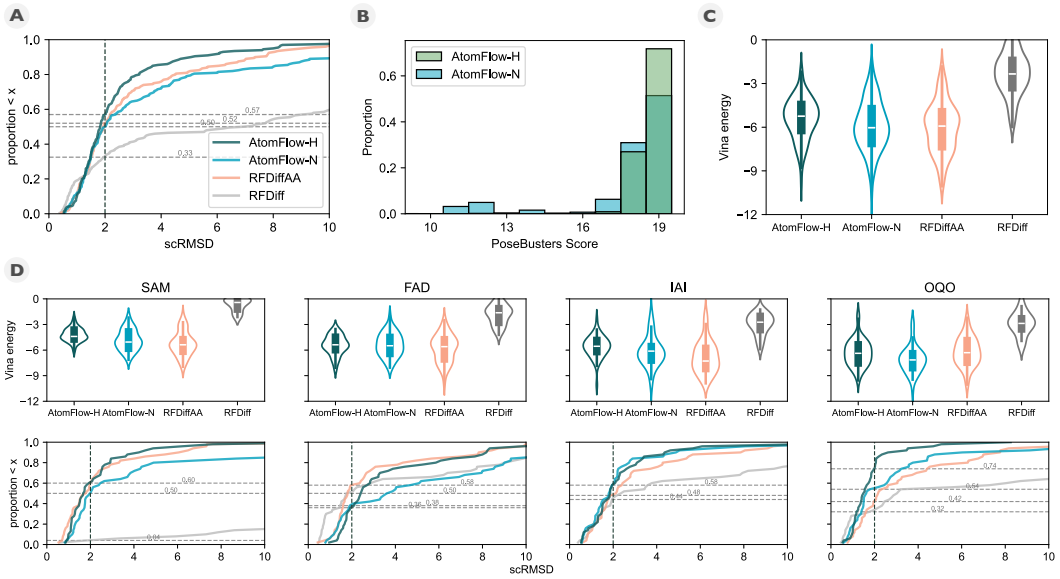

Figure 3: **(A)** scRMSD distribution over the evaluation set. ATOMFLOW outperforms RFDiffusion with a curve similar to RFDiffusionAA. **(B)** PoseBusters score distribution of ATOMFLOW-N on the extended set, showing few violations. **(C)** Vina score distribution (lower is better) on the evaluation set. ATOMFLOW and RFDiffusionAA are comparable, outperforming RFDiffusion. **(D)** Per-ligand results of scRMSD and Vina energy.

5.3 BINDING AFFINITY

We next assess how strongly each designed binder interacts with its ligand via AutoDock Vina Score (Eberhardt et al., 2021), averaged over 8 Rosetta-packed (Leaver-Fay et al., 2011) variants per design. Figure 3C,D illustrates results for the same evaluation set.

RFDiffusion, which lacks direct protein-ligand modeling, yields poor binding energy. By contrast, ATOMFLOW approaches RFDiffusionAA in most cases. ATOMFLOW-H's provided ligand hint can limit exploring alternative binding states, resulting in slightly higher scores.

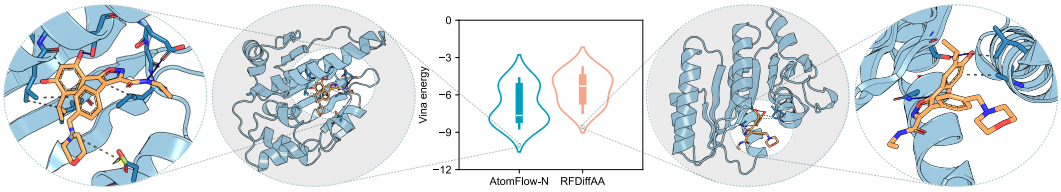

Figure 4: For the 2GJ ligand (luminespib), ATOMFLOW-N outperforms RFDiffusionAA on Vina energy *without* requiring a bound conformer. PLIP (Adasme et al., 2021) analysis suggests that ATOMFLOW's designed binder forms more favorable interactions (right).

We also tested ligand 2GJ (luminespib) where RFDiffusionAA needs the *native* bound structure, but ATOMFLOW does not. ATOMFLOW generated binders with lower energies (Fig. 4), highlighting the benefit of co-modeling flexible conformers.

6 CONCLUSION AND FUTURE WORK

In this work, we proposed ATOMFLOW, a de novo protein binder design method for small molecule ligands considering the flexibility of ligand structure. Unlike previous works, ATOMFLOW no longer relies on a given bound ligand conformer as input. We represent the protein-ligand complex as unified biotokens, learning the structure distribution of both the proteins and the ligands simultaneously from the data with an SE(3)-equivariant flow matching model on the representative atoms.

During the evaluation, ATOMFLOW shows comparable design quality to the state-of-the-art model RFDiffusionAA, which requires the ligand conformer to be fixed before design. Further evaluation exhibits the advantage of ATOMFLOW in the circumstance when the ligand conformer is not known. A direct future work is to support more precise control of the generated structures, and we're working to migrate ATOMFLOW to all kinds of biomolecules, including DNA, RNA, and metal ions.

REFERENCES

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, pp. 1–3, 2024.

Melissa F Adasme, Katja L Linnemann, Sarah Naomi Bolz, Florian Kaiser, Sebastian Salentin, V Joachim Haupt, and Michael Schroeder. Plip 2021: Expanding the scope of the protein–ligand interaction profiler to dna and rna. *Nucleic acids research*, 49(W1):W530–W534, 2021.

Gustaf Ahdritz, Nazim Bouatta, Christina Floristean, Sachin Kadyan, Qinghui Xia, William Gerecke, Timothy J O'Donnell, Daniel Berenberg, Ian Fisk, Niccolò Zanichelli, et al. Openfold: Retraining alphafold2 yields new insights into its learning mechanisms and capacity for generalization. *Nature Methods*, pp. 1–11, 2024.

Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. *arXiv preprint arXiv:2209.15571*, 2022.

Linna An, Meerit Y Said, Long Tran, Sagardip Majumder, Inna Goreshnik, Gyu Rie Lee, David Juergens, Justas Dauparas, Ivan V. Anishchenko, Brian Coventry, Asim K. Bera, Alex Kang, Paul M. Levine, Valentina Alvarez, Arvind Pillai, Christoffer H Norn, David Feldman, Dmitri Zorine, Derrick R. Hicks, Xinting Li, Mariana Garcia Sanchez, Dionne K. Vafeados, Patrick J. Salveson, Anastassia A. Vorobieva, and David Baker. De novo design of diverse small molecule binders and sensors using shape complementary pseudocycles. *bioRxiv*, 2023. URL `https://api.semanticscholar.org/CorpusID:266540105`.

Nathaniel R. Bennett, Brian Coventry, Inna Goreshnik, Buwei Huang, Aza Allen, Dionne Vafeados, Ying Po Peng, Justas Dauparas, Minkyung Baek, Lance Stewart, Frank DiMaio, Steven De Munck, Savvas N. Savvides, and David Baker. Improving de novo protein binder design with deep learning. *Nature Communications*, 14(1):2625, May 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-38328-5.

Helen M. Berman, John D. Westbrook, Zukang Feng, Gary Gilliland, T. N. Bhat, Helge Weissig, Ilya N. Shindyalov, and Philip E. Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28(1): 235–242, 2000.

Matthew J Bick, Per J Greisen, Kevin J Morey, Mauricio S Antunes, David La, Banumathi Sankaran, Luc Reymond, Kai Johnsson, June I Medford, and David Baker. Computational design of environmental sensors for the potent opioid fentanyl. *Elife*, 6:e28909, 2017.

Patrick Bryant, Atharva Kelkar, Andrea Guljas, Cecilia Clementi, and Frank Noé. Structure prediction of protein-ligand complexes from sequence information with umol. *Nature Communications*, 15 (1):4536, 2024.

John-Marc Chandonia, Lindsey Guan, Shiangyi Lin, Changhua Yu, Naomi K Fox, and Steven E Brenner. Scope: improvements to the structural classification of proteins – extended database to facilitate variant interpretation and machine learning. *Nucleic Acids Research*, 50(D1):D553–D559, 2022.

Ricky TQ Chen and Yaron Lipman. Flow matching on general geometries. In *The Twelfth International Conference on Learning Representations*, 2024.

Justas Dauparas, Gyu Rie Lee, Robert Pecoraro, Linna An, Ivan Anishchenko, Cameron Glasscock, and David Baker. Atomic context-conditioned protein sequence design using ligandmpnn. *Biorxiv*, pp. 2023–12, 2023.

Willem Diepeveen, Carlos Esteve-Yagüe, Jan Lellmann, Ozan Öktem, and Carola-Bibiane Schönlieb. Riemannian geometry for efficient analysis of protein dynamics data. *Proceedings of the National Academy of Sciences*, 121(33):e2318951121, 2024.

Jerome Eberhardt, Diogo Santos-Martins, Andreas F. Tillack, and Stefano Forli. Autodock Vina 1.2.0: New Docking Methods, Expanded Force Field, and Python Bindings. *Journal of Chemical Information and Modeling*, 61(8):3891–3898, 2021.

Alex Herbert. Maxcluster: A tool for protein structure comparison and clustering, 2008. URL `http://www.sbg.bio.ic.ac.uk/~maxcluster/`.

John B Ingraham, Max Baranov, Zak Costello, Karl W Barber, Wujie Wang, Ahmed Ismail, Vincent Frappier, Dana M Lord, Christopher Ng-Thow-Hing, Erik R Van Vlack, et al. Illuminating protein space with a programmable generative model. *Nature*, 623(7989):1070–1078, 2023.

Dick Jardine. Euler's method in euler's words. *Mathematical Time Capsules: Historical Modules for the Mathematics Classroom*, (77):215, 2011.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

Michel van Kempen, Stephanie S. Kim, Charlotte Tumescheit, Milot Mirdita, Jeongjae Lee, Cameron L. M. Gilchrist, Johannes Söding, and Martin Steinegger. Fast and accurate protein structure search with Foldseek. *Nature Biotechnology*, 42(2):243–246, 2024.

Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Xiangzhe Kong, Wenbing Huang, and Yang Liu. End-to-end full-atom antibody design. *arXiv preprint arXiv:2302.00203*, 2023.

Rohith Krishna, Jue Wang, Woody Ahern, Pascal Sturmfels, Preetham Venkatesh, Indrek Kalvet, Gyu Rie Lee, Felix S Morey-Burrows, Ivan Anishchenko, Ian R Humphreys, et al. Generalized biomolecular modeling and design with rosettafold all-atom. *Science*, 384(6693):eadl2528, 2024.

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.

Yeqing Lin and Mohammed AlQuraishi. Generating novel, designable, and diverse protein structures by equivariantly diffusing oriented residue clouds. *arXiv preprint arXiv:2301.12485*, 2023.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Zhihai Liu, Minyi Su, Li Han, Jie Liu, Qifan Yang, Yan Li, and Renxiao Wang. Forging the Basis for Developing Protein–Ligand Interaction Scoring Functions. *Accounts of Chemical Research*, 50(2): 302–309, 2017.

Lei Lu, Xuxu Gou, Sophia K Tan, Samuel I Mann, Hyunjun Yang, Xiaofang Zhong, Dimitrios Gazgalis, Jesús Valdiviezo, Hyunil Jo, Yibing Wu, et al. De novo design of drug-binding proteins with predictable binding energy and specificity. *Science*, 384(6691):106–112, 2024.

David L Mobley and Ken A Dill. Binding of small-molecule ligands to proteins:"what you see" is not always "what you get". *Structure*, 17(4):489–498, 2009.

Nicholas F Polizzi and William F DeGrado. A defined structural unit enables de novo design of small-molecule–binding proteins. *Science*, 369(6508):1227–1233, 2020.

Nadine Schneider, Roger A Sayle, and Gregory A Landrum. Get your atoms in order—an open-source implementation of a novel and robust molecular canonicalization algorithm. *Journal of chemical information and modeling*, 55(10):2111–2120, 2015.

Bettina Schreier, Christian Stumpp, Silke Wiesner, and Birte Höcker. Computational design of ligand binding is not a solved problem. *Proceedings of the National Academy of Sciences*, 106(44): 18491–18496, November 2009. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.0907950106.

Chence Shi, Chuanrui Wang, Jiarui Lu, Bozitao Zhong, and Jian Tang. Protein sequence and structure co-design with equivariant translation. *arXiv preprint arXiv:2210.08761*, 2022.

Hannes Stark, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Harmonic Self-Conditioned Flow Matching for joint Multi-Ligand Docking and Binding Site Design. In *Forty-first International Conference on Machine Learning*, volume abs/2310.05764, 2024.

Joseph L. Watson, David Juergens, Nathaniel R. Bennett, Brian L. Trippe, Jason Yim, Helen E. Eisenach, Woody Ahern, Andrew J. Borst, Robert J. Ragotte, Lukas F. Milles, Basile I. M. Wicky, Nikita Hanikel, Samuel J. Pellock, Alexis Courbet, William Sheffler, Jue Wang, Preetham Venkatesh, Isaac Sappington, Susana Vázquez Torres, Anna Lauko, Valentin De Bortoli, Emile Mathieu, Sergey Ovchinnikov, Regina Barzilay, Tommi S. Jaakkola, Frank DiMaio, Minkyung Baek, and David Baker. De novo design of protein structure and function with RFdiffusion. *Nature*, 620(7976):1089–1100, 2023.

Kevin E Wu, Kevin K Yang, Rianne van den Berg, Sarah Alamdari, James Y Zou, Alex X Lu, and Ava P Amini. Protein structure generation via folding diffusion. *Nature communications*, 15(1): 1059, 2024.

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*, 2023a.

Jason Yim, Brian L Trippe, Valentin De Bortoli, Emile Mathieu, Arnaud Doucet, Regina Barzilay, and Tommi Jaakkola. Se (3) diffusion model with application to protein backbone generation. *arXiv preprint arXiv:2302.02277*, 2023b.

Y. Zhang. Tm-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Research*, 33(7):2302–2309, 2005.

Zaixi Zhang, Wanxiang Shen, Qi Liu, and Marinka Zitnik. Pocketgen: Generating Full-Atom Ligand-Binding Protein Pockets. *bioRxiv*, 2024.

A APPENDIX

A.1 PROTEIN FRAMES

Proteins are composed of amino acid chains linked by peptide bonds, forming a backbone with protruding side chains. Each amino acid's position and orientation are described by a local coordinate system, or protein frame, centered on three key backbone atoms: the alpha carbon (Cα), the carbonyl carbon (C), and the amide nitrogen (N). These atoms act as reference points for establishing the frame. The alpha carbon (Cα) typically acts as the origin. The vector from Cα to the amide nitrogen (N) is normalized to define one axis of the frame. A second axis is defined by the normalized vector from Cα to the carbonyl carbon (C). The third axis is formed by the cross product of these two vectors, creating an orthogonal, right-handed coordinate system. The residue frame is typically represented as an SE(3) transformation $T = (R, t)$, which maps a vector from this local system to the global coordinate system. In this transformation, t corresponds to the position of Cα in the global system, and R represents the rotation needed to align the residue's structure within the global context.

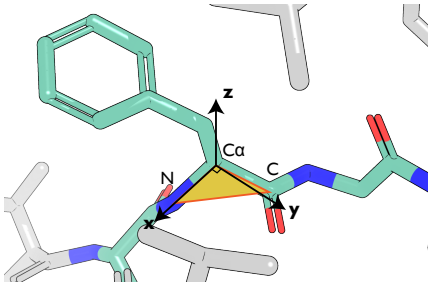

Figure 5: A protein frame illustration. The Cα, C, N atoms form a panel, which is the xy panel. The x-axis is defined as the orientation from Cα to N, while the y-axis is on the panel and perpendicular to the x-axis. The z-axis is perpendicular to the xy panel.

A.2 FLOW MATCHING

Building upon the significant success of diffusion models in various generative tasks, flow matching models (Albergo & Vanden-Eijnden, 2022; Liu et al., 2022) allow for faster and more reliable sampling from a distribution learned from data. The generative process of flow matching models is usually defined by a probability path $p_t(x), t \in [0, 1]$ that gradually transforms from a known noisy distribution $p_0(x) = q(x)$, such as $\mathcal{N}(x|0, I)$ for $x \in \mathbb{R}$, to an approximate data distribution $p_1 \approx p_{\text{data}}(x)$. A vector field $u_t(x)$, which leads to an ODE $\frac{\mathrm{d}\phi_t(\mathbf{x})}{\mathrm{d}t} = u_t(\phi_t(\mathbf{x}))$, is used to generate the probability path via the push-forward equation,

$$p_t = [\phi_t]_* p_0 = p_0(\phi_t^{-1}(x))\det\left[\frac{\partial \phi_t^{-1}}{\partial x}(x)\right], \tag{3}$$

which could be approximated with a trainable network $\hat{v}_t(x; \theta)$.

Due to the complexity of defining an appropriate p_t and u_t, we could alternatively define a conditional probability path $p_t(x|x_1)$, which is usually derived through a conditional vector field $u_t(x|x_1)$ for each data point x_1 (Lipman et al., 2022). The conditional vector field is then approximated with a trainable network $\hat{v}_t(x; \theta)$. Lipman et al. (2022) has proved that the conditional flow matching loss,

$$\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{t, p_{\text{data}}(x_1), p_t(x|x_1)} \|\hat{v}_t(x; \theta) - u_t(x|x_1)\|, \tag{4}$$

has identical gradients w.r.t. θ with $\mathcal{L}_{\text{FM}} = \mathbb{E}_{t, p_{\text{data}}(x)} ||\hat{v}_t(x; \theta) - u_t(x)||$, which means the model can generate a marginal vector field by simply learning from the x_1-conditioned vector fields, without access to $p_t(x)$ and $u_t(x)$. After training, a neural ODE is obtained, ready for sampling from p_0 to p_t with an ODE solver (Jardine, 2011).

A.3 DETAILS ON BIOTOKENS

Token Features For ligand atom tokens, the token-level feature set includes: chirality, degree, formal charge, implicit valence, number of H atoms, number of radical electrons, orbital hybridization,

aromaticity, and ring size. The pair-level feature is provided as one-hot embedding of the bond type. For residue tokens, no token-level feature is known, while the pair-level feature only contains the binned distance of residue index between residues. All features are encoded as a one-hot vector and concatenated.

Token Frames The final loss we adopted $\mathcal{L}_{\text{CFM-FAPE}}$ requires aligning the predicted structure to the local frame of every token. The frames of protein residues can be naturally defined as in Section 3. However, the frames of ligand atoms could not be chosen directly. Since a frame could be calculated from the coordinate of 3 atoms, we need to choose an atom triplet for every atom token.

We first obtain a canonical rank of every atom that does not depend on the input order (Schneider et al., 2015). The atoms are then renamed to their rank. For atoms x with a degree greater than or equal to 2, we select the lexicographically smallest triplet (u, x, v) to define the frame, where u and v are neighbors of x. For atoms with a degree of 1, u is the only neighbor of x, and v is chosen as one of u 's neighbors. This method ensures that each atom's frame is defined in a consistent manner, irrespective of its position in the input sequence, thereby facilitating the model to learn a consistent structural target.

Extending Token Types and Features Though AtomFlow only considers the interaction between protein and molecule ligands, the unified biotoken has the potential to extend to all biological entities, including DNA, RNA, etc, by defining the token position, token frame, local and pair features, and the representation of the internal structure. For example, an RNA can be represented as a sequence of nucleotides, with the token position defined as its mass center, and the token frame calculated from an atom triplet, such as C2-N1-C6.

The token features can also be extended to support more types of known information. For example, the local features could also contain an embedding to indicate the preferred secondary structure, or whether a ligand atom is required to be closer to the designed protein; the pair features could also contain the motif information with a distance map.

A.4 Details on the Flow Matching Process

For all types of tokens, we only consider their token positions to simplify the flow matching process. Thus, the positions of all tokens lie in the Euclidean space $\mathbb{R}^{N\times 3}$. Since a complex could be arbitrarily moved or rotated in the coordinate space without changing its structure, we need an algorithm that treats different position series as the same if they could be aligned with an SE(3) translation. Thus, every data point we consider now lies in the quotient space $\mathbb{R}^{N\times 3}/\text{SE(3)}$. This quotient space is proved to be a Riemannian manifold (Diepeveen et al., 2024).

For a Riemannian manifold, the flow matching process could be defined using a premetric (Chen & Lipman, 2024). A premetric $d : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ should satisfy: 1. $d(x, y) \geq 0$ for all $x, y \in \mathcal{M}$; 2. $d(x, y) = 0$ iff $x = y$; 3. $\nabla d(x, y) \neq 0$ iff $x \neq y$.

We define our premetric as the minimum point-wise rooted sum of squared distance (RMSD) among all pairs of possible structures in the original space $\mathbb{R}^{N\times 3}$ for two elements in the quotient space $d(x, y) = \|\text{align}_x(y) - x\|$, which satisfies all three conditions.

Proof. Since the premetric is defined as a norm, it satisfies condition 1 by nature. When $x = y$, the best alignment that aligns y to x could derive the exact same position as x, yielding a zero norm. When $x \neq y$, when y is aligned to x, there's still a structural difference between the structures, thus the premetric is not zero. For condition 3, by defining $y' = \text{align}_x(y)$, we have

$$\nabla d(x, y) = \nabla \sqrt{\sum_{i=1}^{n} (y'_i - x_i)^2} = \frac{y' - x}{||y' - x||} = \frac{\text{align}_x(y) - x}{||\text{align}_x(y) - x||} \geq 0. \tag{5}$$

Thus $d(x, y)$ satisfies all the conditions as a qualified premetric. □

With such premetric, and a monotonically decreasing differentiable scheduler $\kappa(t) = 1 - t$, we could obtain a well-defined conditional vector field that linearly interpolates between the noisy and real

data (Chen & Lipman, 2024)

$$u_t(x|x_1) = \frac{\mathrm{d}\log\kappa(t)}{\mathrm{d}t} d(x, x_1) \frac{\nabla d(x, x_1)}{\|\nabla d(x, x_1)\|^2} = \frac{1}{1-t}(\text{align}_x(x_1) - x). \tag{6}$$

The vector field in equation 6 is calculated by substituting equation 5 into the left side. This vector field provides the direction for moving straight towards x_1, and generates a probability flow that interpolates linearly between noisy sample x_0 and data sample x_1.

Since the vector field is defined as a function of x_1, we could learn the vector field with a structure prediction model $\hat{x_1}(x, t; \theta)$. By substituting equation 6 into equation 4, we obtain the training loss

$$\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{t, p_{\text{data}}(x_1), p_t(x|x_1)} \left\| \frac{1}{1-t}(\text{align}_x(\hat{x_1}(x, t; \theta)) - \text{align}_x(x_1)) \right\|. \tag{7}$$

A.5 Details on the Prediction Network

Structure Module Specifications The main components of the structure module are derived from Alphafold 2 (Jumper et al., 2021), while our implementation builds on top of the widely acknowledged reimplementation OpenFold (Ahdritz et al., 2024). The TransformerStack consists of 14 layers of simplified Evoformer block, and the IPAStack consists of 4 layers of Invariant Point Attention (IPA) blocks. The MSA operations in the Evoformer block are simplified by replacing the operations on the MSA feature matrix with the single representation s_i. The weights of the IPA blocks are shared, and the structural loss is calculated on the outputs of each block and averaged.

Training Details During training, we equally sample data from the SCOPe dataset (v2.08) and the PDBBind dataset (2020). We simply drop the data with more than 512 tokens, and we don't crop the filtered complexes since the cutoff is large enough and only filters out a relatively small portion of the data. We train our model on 10 NIVIDA RTX 4090 acceleration cards, with a batch size set to 10, which means the batch size on each device is set to 1. We use the Adam Optimizer (Kingma, 2014) with a weight-decaying learning rate scheduler, starting from 10^{-3} and decays the learning rate by 0.95 every 50k steps. We separate the training process into two stages: 1) initial training, $\alpha_1 = 0.5, \alpha_4 = 0.3, \alpha_2 = \alpha_3 = 0$; 2) finetuning, $\alpha_1 = \alpha_2 = \alpha_3 = 0.5, \alpha_4 = 0.3$.

Ligand tokens are not given during the first training stage. The first stage trains an unconditional protein generation model, while the second stage turns it to a conditional protein binder and ligand conformer generation model. The FAPE loss is defined as an average of all pairs of tokens in the original paper, so the calculation process first yield a FAPE matrix and then produce the average value of the matrix. The protein-protein, protein-ligand and ligand-ligand loss calculates the average value of the sub-matrixs defined as (row: protein, col: protein), (row: protein, col: ligand), and (row: ligand, col:ligand).

Since training a protein design model is significantly time-consuming, the design choices of our training strategy is largely determined by grid searching possible design space and save the training trajectory of the first 30$\sim$50k steps. We compare the training trajectories and select the best configuration that meets the following criteria: a) The final distogram loss should close the minimum we get among the configurations (around 2.0). b) The $\mathcal{L}_{\text{CFM-FAPE}}$ should not decline too fast at the first 10k steps. The first 10k steps is for the transformer stack to learn a relatively steady output, indicated by the decline of the distogram loss. A decline of $\mathcal{L}_{\text{CFM-FAPE}}$ at this stage will resulted in an undesired local minimum. Then $\mathcal{L}_{\text{CFM-FAPE}}$ should decline fast right after the distogram loss turns to decline much smoother. We select the configuration with the lowest $\mathcal{L}_{\text{CFM-FAPE}}$ at the end of training.

We decide the end of each training stage when the training converges, with the following criteria: a) the decline rate of every single loss is small. b) the structural violence of sampled structures (counts of CA atom violation) converges.

An initial study on directly train the second stage shows unsatisfactory training trajectory. Since the ligand conformer is way easier to generate compared to protein folds, the FAPE loss declines too fast even before the distogram loss, resulted in unstable TransformerStack output, and leading to a diverge of the model after 30k steps. The resulted model with minimum loss is able to predict the ligand structure, with random protein residue position, which is unusable.

Loss Function $\mathcal{L}_{\text{CFM}}$ calculates an aligned RMSD by aligning $\mathbf{x_1}$ and $\hat{\mathbf{x_1}}$ to $\mathbf{x}$, while the FAPE loss calculates an averaged RMSD by aligning $\hat{\mathbf{x_1}}$ to each residue frame of $\mathbf{x_1}$, which could be extended to the token frame (Appendix A.3). Let $\text{align}_{x,i}(y)$ denote aligning y to the i-th token frame of x, we have

$$\begin{aligned}
\mathcal{L}_{\text{CFM}} &= \mathbb{E}_{t,p_{\text{data}}(x_1),p_t(x|x_1)} \left\| \frac{1}{1-t}(\text{align}_x(\hat{x_1}(x,t;\theta)) - \text{align}_x(x_1)) \right\| \\
&\approx \mathbb{E}_{t,p_{\text{data}}(x_1),p_t(x|x_1)} \left\| \frac{1}{1-t} \cdot \frac{1}{N} \sum_{i=1}^{N} \left(\text{align}_{x,i}(\hat{x_1}(x,t;\theta)) - \text{align}_{x,i}(x_1)\right) \right\| \\
&\approx \mathbb{E}_{t,p_{\text{data}}(x_1),p_t(x|x_1)} \left\| \frac{1}{1-t} \cdot \frac{1}{N} \sum_{i=1}^{N} \left(\text{align}_{x_1,i}(\hat{x_1}(x,t;\theta)) - \text{align}_{x_1,i}(x_1)\right) \right\| \\
&\approx \mathbb{E}_{t,p_{\text{data}}(x_1),p_t(x|x_1)} \left\| \frac{1}{1-t} \cdot \frac{1}{N} \sum_{i=1}^{N} \left(\text{align}_{x_1,i}(\hat{x_1}(x,t;\theta)) - x_1\right) \right\| \\
&= \mathcal{L}_{\text{CFM-FAPE}}
\end{aligned}$$

Proposition 1. *align*$_{\mathbf{x_1}}(\hat{\mathbf{x_1}}) = \mathbf{x_1} \iff \mathcal{L}_{\textit{CFM}} = 0 \iff \mathcal{L}_{\textit{CFM-FAPE}} = 0$.

Proof. When $\text{align}_{\mathbf{x_1}}(\hat{\mathbf{x_1}}) = \mathbf{x_1}$, we have $\forall i, \text{align}_{\mathbf{x_1},i}(\hat{\mathbf{x_1}}) = \mathbf{x_1}$. As a result, $\mathcal{L}_{\text{CFM}} = \mathcal{L}_{\text{CFM-FAPE}} = 0$. This establishes that:

$$\text{align}_{\mathbf{x_1}}(\hat{\mathbf{x_1}}) = \mathbf{x_1} \iff \mathcal{L}_{\text{CFM}} = 0 \quad \text{and} \quad \text{align}_{\mathbf{x_1}}(\hat{\mathbf{x_1}}) = \mathbf{x_1} \iff \mathcal{L}_{\text{CFM-FAPE}} = 0. \tag{8}$$

Now, assume $\mathcal{L}_{\text{CFM}} = 0$. Suppose $\text{align}_{\mathbf{x_1}}(\hat{\mathbf{x_1}}) \neq \mathbf{x_1}$. Then for all transformations R and t, we have $R\hat{\mathbf{x_1}} + t \neq \mathbf{x_1}$, which implies: $\|\text{align}_{\mathbf{x_1}}(\hat{\mathbf{x_1}}) - \mathbf{x_1}\| \neq 0$, leading to $\mathcal{L}_{\text{CFM}} \neq 0$. This is a contradiction. Therefore, $\text{align}_{\mathbf{x_1}}(\hat{\mathbf{x_1}}) = \mathbf{x_1}$. This proves that

$$\mathcal{L}_{\text{CFM}} = 0 \iff \text{align}_{\mathbf{x_1}}(\hat{\mathbf{x_1}}) = \mathbf{x_1}. \tag{9}$$

Similarly, assume $\mathcal{L}_{\text{CFM-FAPE}} = 0$. Suppose $\text{align}_{\mathbf{x_1}}(\hat{\mathbf{x_1}}) \neq \mathbf{x_1}$. Then: $\|\text{align}_{\mathbf{x_1},i}(\hat{\mathbf{x_1}}) - \mathbf{x_1}\| \neq 0$, which leads to $\mathcal{L}_{\text{CFM-FAPE}} \neq 0$, again a contradiction. Therefore, $\text{align}_{\mathbf{x_1}}(\hat{\mathbf{x_1}}) = \mathbf{x_1}$. This proves that:

$$\mathcal{L}_{\text{CFM-FAPE}} = 0 \iff \text{align}_{\mathbf{x_1}}(\hat{\mathbf{x_1}}) = \mathbf{x_1}. \tag{10}$$

The proposition is proved by combining equation 8,9,10. □

This means that both $\mathcal{L}_{\text{CFM}}$ and $\mathcal{L}_{\text{CFM-FAPE}}$ provide an optimization direction towards minimizing the SE(3) invariant structural difference between the predicted structure and the ground truth structure. Thus, we adopt $\mathcal{L}_{\text{CFM-FAPE}}$ as a realistic approximation of $\mathcal{L}_{\text{CFM}}$ and adopt it as the training objection during evaluation.

A.6 Evaluation Details

Specifications Following RFDiffusionAA, we use FAD, SAM, IAI, and OQO as the selected evaluation set. FAD and SAM are witnessed by both models as training data, while IAI and OQO are not, demonstrating the generalization ability. To further investigate the performance of our method, we conduct experiments on an extended set of 20 ligands (ligands from PDB id 6cjs, 6e4c, 6gj6, 5zk7, 6qto, 6i78, 6ggd, 6cjj, 6i67, 6iby, 6nw3, 6o5g, 6hlb, 6efk, 6gga, 6mhd, 6i8m, 6s56, 6tel, and 6ffe). The extended dataset includes ligand sizes (including hydrogen) ranging from 21 to 104 in length.

Extended Set We illustrate the designability (scRMSD) and binding affinity (Vina energy) of AtomFlow-N in Figure 7. The extended evaluation shows that the performance of AtomFlow on the extended set is similar to the evaluation set shown in the main article, and demonstrates that AtomFlow is able to tackle almost all kinds of ligands.

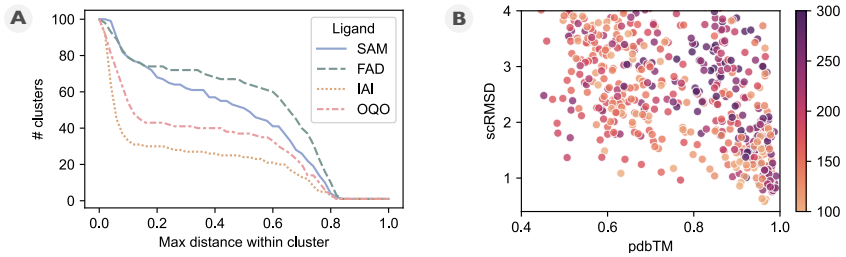

Figure 6: **(A)** Cluster counts for each ligand at various thresholds; ATOMFLOW is consistently diverse. **(B)** Scatter of designability (scRMSD) vs. novelty (pdbTM). ATOMFLOW achieves self-consistency and high novelty, although most designable folds still resemble known proteins.

Diversity and Novelty We measure *diversity* by clustering 100 designed structures (200 residues each) per ligand with MaxCluster (Herbert, 2008), counting how many clusters form at different distance thresholds. We measure *novelty* by the highest TM-score (Zhang, 2005) found via FoldSeek (Kempen et al., 2024) (pdbTM) and each design's scRMSD.

Figure 6A shows that our method produces multiple distinct folds, suggesting effective protein-only training in addition to complexes. Figure 6B indicates that ATOMFLOW can generate novel structures with acceptable designability, though truly new folds remain challenging.

Additional Binding Affinity Metric We are aware that Vina might not be a perfect proxy for binding affinity. We noticed that AlphaProteo adopts several metrics produced by AlphaFold 3 as in silico filters. At the time we finished our draft, there's no publicly available AlphaFold 3 for us to run locally. Recently, several AlphaFold 3 replications and the original AlphaFold 3 are released. We developed an alternative in silico metric based on Chai-1, calculating the minimum value across all interchain terms in the PAE matrix (`min_ipAE`, lower is better). Note that this metric is proved to be a good indicator for protein-protein binder design, but not validated on small molecule-protein binders. The results in Figure 8 show that the AtomFlow-generated binders have lower `min_ipAE` than the ones from RFDiffusion-AA, and both models have the ability to generate binders with similar `min_ipAE` as natural ones. We're working to develop better metrics as in silico proxy for ligand-binding protein design based on wet lab verification.

Diversity and Novelty Results of the Baseline. We conducted the diversity and novelty experiment on RFDiffusion-AA with the same configuration as our results reported in the main text. The results are shown in Figure 9. The diversity of AtomFlow designs is better than RFDiffusion-AA, while the AtomFlow generated results tends to be more conservative in terms of pdbTM novelty. We believe this is because we didn't train AtomFlow on a full training set including all PDB structures and the distillation data. This is our future work and we'll release an updated model once available.

Discussion on Pocket Design Models While the pocket design models address ligand-protein interactions, their focus is limited to refining pocket residues within a predefined radius. They lack the capacity to design full protein folds, making direct comparison with AtomFlow infeasible. We conducted an unfair experiment with PocketGen by providing a template binder to it, as detailed in Tabel 1. Despite this, the results demonstrate that AtomFlow consistently outperforms PocketGen in terms of fold quality across all radii.

Geometrical Distributions of Generated Structure We evaluated the common chemical bond length generated by AtomFlow vs. the ground truth bond length in our training set. Results shown in Figure 10 demonstrate that the AtomFlow generated ligands have similar geometric distribution to ground truth. We further evaluated the generated structures by plotting the Ramachandran plots. Results shown in Figure 11 suggests that the proteins generated by AtomFlow effectively capture the key structural characteristics of natural proteins.

Chemical Validity We evaluated the generated ligands with several important chemical validity metrics: QED, an index of drug-likeness, with a value between 0 (drug-unlike) and 1 (drug-like); SA,

the difficulty of chemical synthesis for molecules, with a value between 0 (easy to synthesize) and 10 (very difficult to synthesize); LogP, an important parameter to characterize the overall hydrophobicity of organic compounds. Results are shown in Tabel 2.

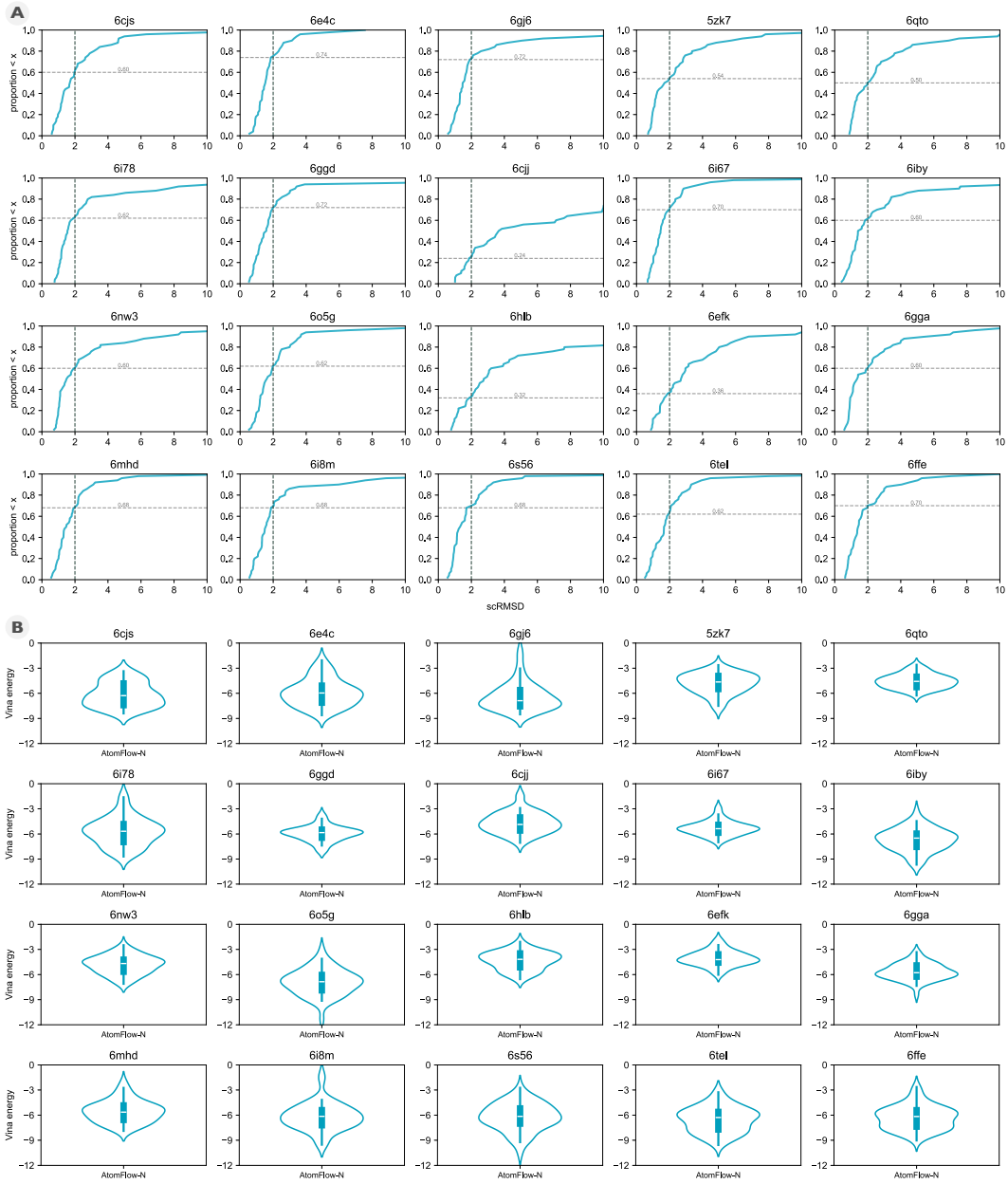

Figure 7: A: scRMSD of designs for each ligand in the extended set; B: Vina energy of designs for each ligand in the extended set.

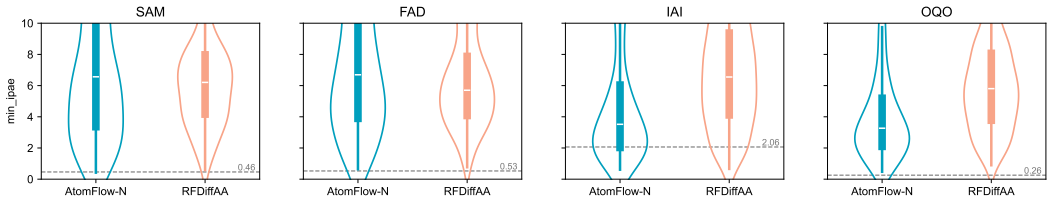

Figure 8: `min_ipAE` distribution of the generation results in the affinity experiment of the main text. The result of the native binder is displayed as a grey line.

Ligand	AtomFlow (r=inf)	PG (r=3.5)	PG (r=5)	PG (r=6.5)	PG (r=8)	PG (r=9.5)
FAD	0.79/3.74	7.10/7.38	6.75/7.81	7.29/8.35	20.92/24.23	23.12/25.23
SAM	0.83/2.01	2.12/2.62	2.77/2.99	2.94/4.03	12.39/14.49	13.79/14.74
IAI	0.56/1.82	0.71/0.85	0.95/1.02	2.04/2.28	3.59/5.53	9.02/11.71
OQO	0.59/1.63	1.20/1.26	1.70/1.79	2.40/2.45	11.59/11.94	2.13/2.41

Table 1: For this experiment, we used the natural binders of four ligands—FAD (7bkc), SAM (7c7m), IAI (5sdv), and OQO (7v11)—as input. To evaluate the design capability of PocketGen (PG) under different constraints, we progressively increased the design radius (minimum distance to ligand) from 3.5 to 9.5. The masked target area expanded with the radius, requiring the model to redesign increasingly larger regions of the protein. When the radius exceeded the protein's dimensions (radius greater than the protein size), all residues were masked, simulating our full-design setting. The table below presents the min/median scRMSD values for designs generated by PocketGen at each radius. For reference, scRMSD < 2 is generally considered a successful design. Notably, PocketGen's performance deteriorated significantly as the radius increased, reflecting its reliance on template residues. (At radius=8 for OQO, PocketGen generated designs with several residues misaligned with the ligand, leading to abnormally high scRMSD values.) Additionally, PocketGen does not support radius settings beyond 10, preventing direct simulation of AtomFlow 's fully template-free design scenario. The results of AtomFlow is derived from our main experiment.

	QED	SA	LogP
AtomFlow	0.458±0.292	0.683±0.160	0.559±4.030
PDBBind	0.429±0.246	0.696±0.144	-0.311±3.240

Table 2: QED, SA, and LogP of AtomFlow generated structures and PDBBind structures.

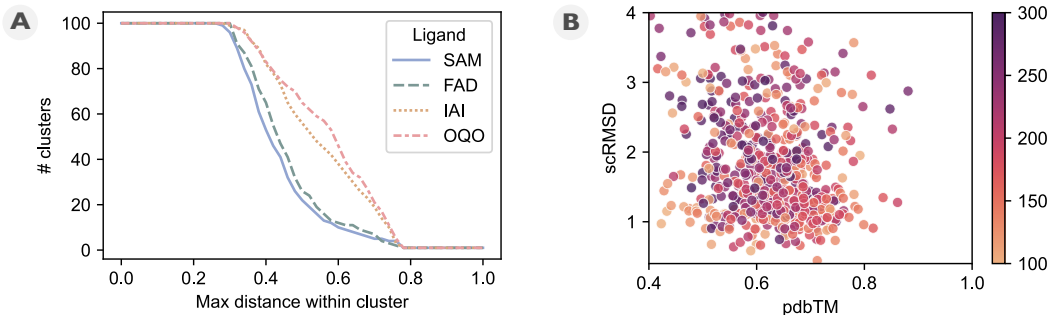

Figure 9: **A**: Cluster count based on different thresholds for the maximum difference within the cluster for each ligand in the evaluation set. **B**: Scatter plot of designability (scRMSD) vs. novelty (pdbTM) for ligands in the evaluation set.

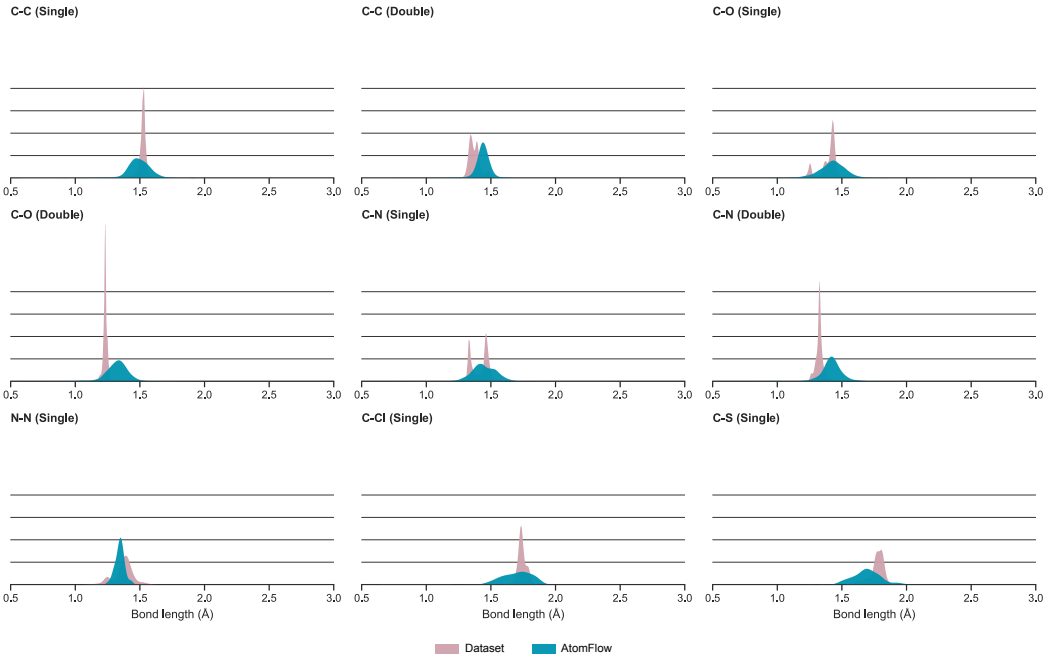

Figure 10: Chemical bond distribution of AtomFlow generated ligands for the extended set and ground truth ligands in the PDBBind dataset.

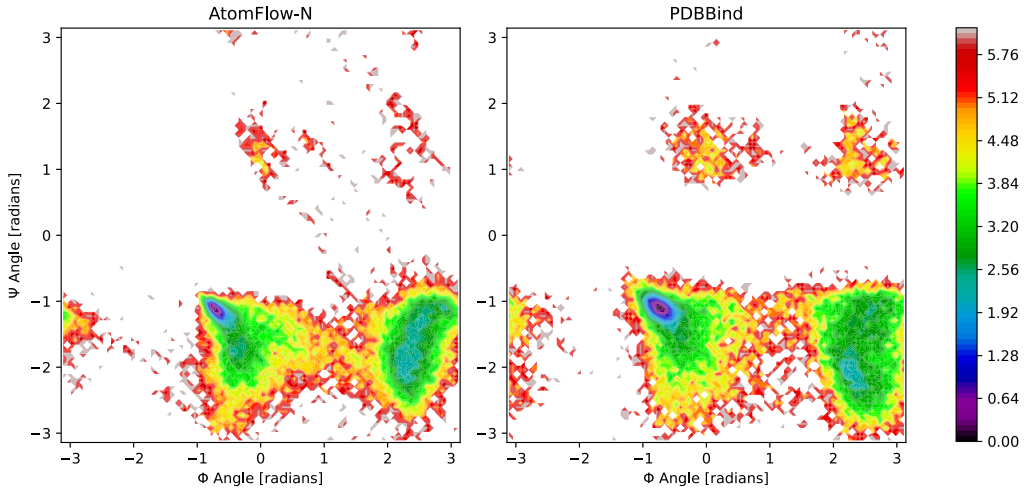

Figure 11: The Ramachandran plots for the generated protein (left) and the PDBBind protein (right), which demonstrate comparable **coverage** in the primary secondary structure regions.