PocketX: Preference Alignment for Protein Pockets Design through Group Relative Policy Optimization

Yuliang Fan[1*], **Zaikai He**[1*], **Bin Li**[6], **Bin He**[6], **Mingshu Zhang**[7,8†], **Jian Zhang**[1,2,3,5†], and **Haicang Zhang**[1,2,3,4†]

[1]Department of Pharmaceutical and Artificial-Intelligence Sciences, Shanghai Jiao Tong University School of Medicine
[2]Shanghai Key Laboratory of Flexible Medical Robotics, Tongren Hospital, Institute of Medical Robotics, Shanghai Jiao Tong University
[3]Artificial Intelligence Clinical Research Center for drug discovery, Tongren Hospital, Shanghai Jiao Tong University School of Medicine
[4]Central China Research Institute of Artificial Intelligence
[5]College of Pharmacy, Ningxia Medical University, Yinchuan, China
[6]Institute of Medical Technology, Peking University
[7]Neuroscience Research Institute, Peking University
[8]Institute of Advanced Clinical Medicine, Peking University

Abstract

Designing protein pockets that target specific ligands is crucial for drug discovery and enzyme engineering. Although deep generative models show promise in proposing high-quality pockets, they are usually trained purely to match the data distribution and therefore overlook key biophysical properties, such as binding affinity, expression, and solubility, that ultimately determine developability and success. We introduce PocketX, an online reinforcement learning framework that explicitly aligns a generative model with desired biophysical properties. The framework first trains a base model that co-designs pocket structures and sequences conditioned on a target ligand, and then fine-tunes this model with Group Relative Policy Optimization (GRPO) to reward the desired attributes. Because GRPO employs group-relative rewards, it produces lower-variance policy updates, resulting in more stable and efficient learning than competing alignment strategies. Evaluated on the CrossDocked2020 benchmark, PocketX surpasses existing methods in metrics such as binding energy and evolutionary plausibility. Ablation studies further show that GRPO outperforms alternative alignment strategies, including Direct Preference Optimization (DPO), confirming GRPO's effectiveness for biophysical property alignment.

1 Introduction

Designing protein pockets that bind specific ligands is fundamental to advances in drug discovery [37], clinical diagnostics [26], enzyme engineering [5], and biosensor development [44]. Existing computational approaches fall into two main categories: physics-based and template-based. Physics-based algorithms, such as PocketOptimizer [20, 35], perform combinatorial searches over sequence space to minimize calculated binding free energy. Template-based methods [3, 45, 25, 18] construct pockets by assembling structural motifs around the target ligand while enforcing specific hydrogen-

[*]These authors contributed equally.
[†]Corresponding authors: zhanghaicang@sjtu.edu.cn, jian.zhang@sjtu.edu.cn, mszhang@hsc.pku.edu.cn

Machine Learning for Structural Biology Workshop, NeurIPS 2025.

bonding patterns. Despite their successes in particular cases, both strategies are constrained by limited accuracy and substantial computational cost.

Recent breakthroughs in deep generative modeling for languages [39, 21] and images [33, 29] are now reshaping protein pocket design. For example, FAIR [46] and PocketGen [47] utilize graph-based neural networks to predict the pocket structures and sequences directly, whereas RFdiffusion [42] and its variant RFdiffusion All-Atom (RFdiffusionAA) [14] leverage diffusion-based generative models to sample the pocket sequence and structures. Because these methods are trained primarily to reproduce the statistical distribution of existing data, they often neglect critical biophysical properties—such as binding affinity, expression level, and solubility—that ultimately determine developability and success in practice [24, 7].

To move beyond simple distribution matching, reinforcement learning (RL) provides a principled way to steer generation toward task-specific objectives. In molecular design, Direct Preference Optimization (DPO)–based methods have already improved antibody and small-molecule design [49, 28, 10] with more desired properties such as binding energy and stability. More recently, DeepSeek's Group Relative Policy Optimization (GRPO) [11, 32]—an online RL algorithm originally developed for autoregressive language models—has been shown to surpass DPO and PPO [31] in both training stability and sample efficiency [11]. GRPO [11, 32] has since been generalized beyond autoregressive language modeling to diffusion- and flow-based generative models such as DanceGRPO [43] and FlowGRPO [16] and also achieves similar performance gains.

Motivated by these advances, we propose PocketX, a deep generative model for protein pocket design that leverages GRPO to steer the designed pockets toward desired biophysical properties. PocketX first trains a hybrid continuous–discrete diffusion model for pocket structure and sequence co-generation, using AlphaFold3-style architectures as the score network [1]. The resulting base model is then fine-tuned with GRPO. Physics-based rewards, such as AutoDock Vina docking scores [38, 8], promote high binding affinity, while model-based rewards, such as ESM-2 sequence perplexity [15], encourage evolutionary plausibility. Unlike DPO's pairwise preference alignment, PocketX treats these metrics as continuous reward signals, providing richer feedback and finer control in an online RL setting.

Our key contributions are summarized as follows:

- We propose PocketX, a diffusion-based generative model for pocket design that incorporates GRPO to steer generation with biophysical constraints. To the best of our knowledge, this is the first application of GRPO to the diffusion-based generative models for protein design.
- Experiments show that PocketX achieves state-of-the-art performance in generating pockets with more desired properties, such as binding affinity and evolutionary plausibility.

2 Methods

In this section, we introduce PocketX, consisting of two training phases, as shown in Figure 1. First, we train a ligand-conditioned generative model [36] to co-design pocket structures and sequences, referred to as the base model (see the section A in the appendix for more details). Next, we fine-tune the base model using the RL algorithm GRPO, where the optimized network learns a policy that simultaneously accounts for biophysical energy and evolutionary plausibility preference. Section 2.1 outlines the preliminaries and notations, and Section 2.2 details the GRPO framework for protein pocket generation.

2.1 Problem Formulation

Following the convention in previous works [46, 47], pocket generation in PocketX is formulated as a conditional generation problem that generates the sequence and structure of the pocket conditioned on the target ligand and the protein scaffold (the protein regions outside the pocket). We model protein–ligand complex as $\mathcal{C} = \{\mathcal{P}, \mathcal{M}, \mathcal{S}\}$, consisting of the protein pocket $\mathcal{P}$, protein scaffold $\mathcal{S}$ and the ligand molecule $\mathcal{M}$. The ligand molecule can be represented as a 3D point cloud $\mathcal{M} = \{(\mathbf{x}_M^{(i)}, \mathbf{v}_M^{(i)})\}_{i=1}^{N_M}$, where $\mathbf{x}_M \in \mathbb{R}^3$ and $\mathbf{v}_M \in \mathbb{R}^K$ denote the atomic 3D coordinates and atom type respectively, and K being the size of the atom type vocabulary and N_M the number of atoms in the ligand molecule. The protein pocket and protein scaffold are represented as a sequence of

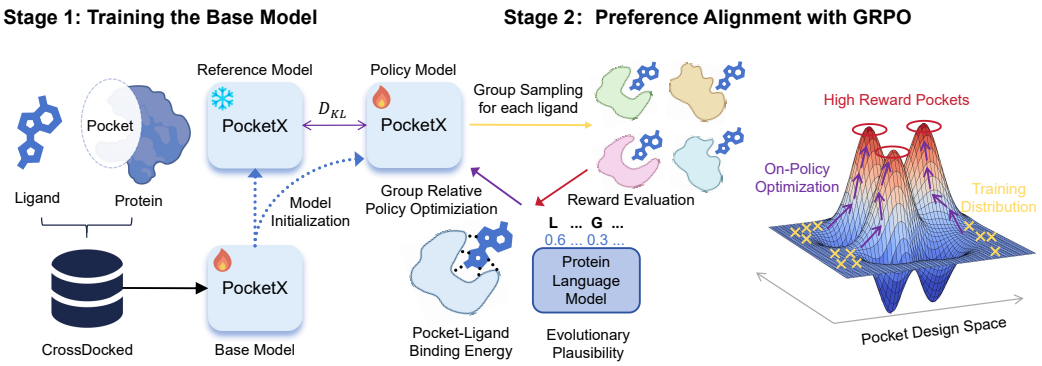

Figure 1: Overview of PocketX. A two-stage framework where a ligand-conditioned base model is trained for pocket structure–sequence co-design, then finetuned with GRPO using biophysical and evolutionary rewards. A reference model regularizes training to avoid over-optimization.

residues $\mathcal{P} = \{(\mathbf{x}_P^{(i)}, \mathbf{v}_P^{(i)})\}_{i=1}^{N_P}$ and $\mathcal{S} = \{(\mathbf{x}_S^{(i)}, \mathbf{v}_S^{(i)})\}_{i=1}^{N_S}$, respectively, where N_P and N_S denote the number of amino acids in the protein pocket $\mathcal{P}$ and the scaffold $\mathcal{S}$. Note that we represent the 3D atomic positions of a protein residue as $\mathbf{x} \in \mathbb{R}^{14\times 3}$, where 14 is the largest number of atoms of any possible amino acid in the designed protein pocket. With the preceding notations, PocketX is defined as a conditional generative model, formally expressed as $p_\theta(\mathcal{P} \mid \mathcal{S}, \mathcal{M})$.

2.2 Group Relative Policy Optimization for Pocket Design

To steer PocketX toward desired biophysical properties, we adopt GRPO [11, 32]. Originally developed for autoregressive language modeling, GRPO's group-relative updates rely solely on scalar rewards, making it model-agnostic and readily applicable to diffusion and flow models [43, 16]. We apply it to our denoising policy, treating each diffusion trajectory as a rollout and optimizing continuous physics- and model-based rewards online. For each ligand $\mathcal{M}$, the generative model samples a group of G individual pockets $\{\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_G\}$ from the old policy $\pi_{\theta_{\text{old}}}$ and then optimizes the current policy model π_θ with the training objective:

$$\max_\theta \ \mathcal{J}(p_\theta) = \mathbb{E}_{m\sim\mathcal{M},\{\mathcal{P}_i\}_{i=1}^G\sim\pi_{\theta_{\text{old}}}(\cdot|m)} \frac{1}{G}\sum_{i=1}^{G}\frac{1}{T}\sum_{t=0}^{T-1}\left(\min\left(\rho_{t,i}(\theta)\hat{A}_i^t,\ \text{clip}(\rho_{t,i}(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_i^t\right) - \beta D_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{ref}})\right), \tag{1}$$

$$\rho_{t,i}(\theta) = \frac{\pi_\theta(\mathcal{P}_i^{t-1} \mid \mathcal{P}_i^t, m)}{\pi_{\theta_{\text{old}}}(\mathcal{P}_i^{t-1} \mid \mathcal{P}_i^t, m)}, \quad \hat{A}_i^t = \frac{R(m,\mathcal{P}_i) - \text{mean}(\{R(m,\mathcal{P}_i)\}_{i=1}^G)}{\text{std}(\{R(m,\mathcal{P}_i)\}_{i=1}^G)}, \tag{2}$$

where t is the noise level of diffusion process, $\rho_{t,i}(\theta)$ is the probability ratio, π_{ref} is a reference policy, ϵ and β are hyperparameters, and $\hat{A}_i^t$ is the advantage, computed using a group of rewards $\{R(m,\mathcal{P}_i)\}_{i=1}^G$ corresponding to the outputs in each group. $D_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{ref}})$ is the KL divergence between the updated policy and the reference policy. $\pi_\theta(\mathcal{P}_i^{t-1} \mid \mathcal{P}_i^t, m)$ stands for the denoising step, also known as the reverse process:

$$\pi_\theta(\mathcal{P}^{t-1} \mid \mathcal{P}^t, m) = \mathcal{N}\left(\mathbf{x}_P^{t-1}; \mu_\theta([\mathbf{x}_P^t, \mathbf{v}_P^t], t, \mathbf{m}), \tilde{\beta}_t I\right) \cdot \mathcal{C}\left(\mathbf{v}_{t-1}; c_\theta([\mathbf{x}_P^t, \mathbf{v}_P^t]], t, \mathbf{p})\right), \tag{3}$$

where $\mathcal{N}$ and $\mathcal{C}$ denote the Gaussian and categorical distributions, respectively, with their parameters approximated by the model's predictions μ_θ and c_θ.

2.2.1 Reward Design

The whole point of RL is to find a policy that maximizes the expected cumulative reward. Therefore, the reward serves as the training signal of RL, determining the optimization direction [11]. Recognizing the superiority of evolutionary plausibility in protein design [12, 50, 28] and binding energy in

drug design [10], we integrate them as reward functions in our RL framework, defined as a weighted combination of evolutionary plausibility and binding energy:

$$R = w_{\text{evo}} R_{\text{evo}} + w_{\text{bind}} R_{\text{bind}}, \tag{4}$$

where R_{evo} and R_{bind} are normalized reward components, and we empirically set their respective weights $w_{\text{evo}} = 0.5$ and $w_{\text{bind}} = 1.0$.

Evolutionary plausibility reward Evolutionary Plausibility measures how likely a designed sequence is evolutionarily plausible in nature, reflecting adherence to general evolutionary rules of natural [12]. It is evaluated using the likelihood under an independent protein language model. Specifically, we evaluate evolutionary plausibility by calculating the perplexity of the protein language model for the designed pocket region. Here, we use $\{\mathbf{v}_P^{(i)}\}_{i=1}^{N_P}$ to represent the residue types of the designed pocket $\mathcal{P}$. The specific formula is formally described as follows:

$$\text{Evolutionary Plausibility} = \sum_{i=1}^{N_P} -\log P(\mathbf{v}_P^{(i)} \mid \mathcal{S}, \mathcal{P} \setminus \mathbf{v}_P^{(i)}). \tag{5}$$

Binding energy reward The binding affinity between the ligand and the receptor protein was evaluated using AutoDock Vina. The Vina scoring function is an optimized, empirical function that provides an estimate of the binding free energy for the ligand-receptor complex. It decomposes the intricate network of intermolecular interactions into a weighted sum of computationally tractable terms. The general form of the scoring function is expressed as:

$$\text{Vina Energy} = \sum_{i}^{\text{ligand}} \sum_{j}^{\text{protein}} [E_{\text{vdW}}(r_{ij}) + E_{\text{H-bond}}(r_{ij}) + E_{\text{Hydrophobic}}(r_{ij})] + E_{\text{internal}} + \Delta G_{\text{tors}}, \tag{6}$$

where i and j index the atoms of the ligand and the protein, respectively. r_{ij} is the distance between atoms i and j. $E_{\text{vdW}}(r_{ij})$, $E_{\text{H-bond}}(r_{ij})$, $E_{\text{Hydrophobic}}(r_{ij})$, and E_{internal} represent the van der Waals interaction energy, hydrogen bonding energy, hydrophobic free energy, and the internal energy of the ligand, respectively. $\Delta G_{\text{tors}} = w_{\text{tors}} \cdot N_{\text{tors}}$ is a torsion penalty proportional to the number of rotatable bonds in the ligand (N_{tors}) restricted upon binding.

3 Results

Datasets. We train and evaluate PocketX on CrossDocked2020 [9], which comprises 22.5M cross-docked protein–molecule pairs. Following prior work [46, 47], we remove samples with binding pose RMSD > 1 Å, yielding $\sim$180k data points. For splitting, sequences are clustered at 30% identity using MMseqs2 [34], from which we select 75k pairs for pre-training, 15k pairs for reinforcement learning fine-tuning, and 100 pairs from the remaining clusters for validation and testing. Consistent with previous work [46, 47], the pocket region is defined as the set of all protein residues containing atoms within 3.5 Åof any ligand atom, following the conventional distance ranges pertaining to protein-ligand interactions [19]. Evaluation is performed based on 100 independently sampled pockets for each ligand in the test set. More details about implementation can be seen in section B.

Evaluation Metrics. We adopt the same evaluation metrics as prior work [46, 47] to assess both the sequence and structural validity of generated pockets. Amino Acid Recovery (**AAR**) measures the sequence recovery accuracy by comparing the generated sequences to the native sequences. Self-consistency Root Mean Squared Deviation (**scRMSD**) measures the deviation between the generated backbone atoms and the predicted pocket's backbone, serving as an indicator of structural plausibility. Specifically, for each generated protein structure, eight sequences are derived by ProteinMPNN [4] and then folded to structures with ESMFold [15]. Binding energy is evaluated with **AutoDock Vina** [8] and **GlideSP**, whereas evolutionary plausibility is measured by the likelihood under an independent protein language model **ESM-2** [15].

Baseline Methods. We evaluate PocketX in comparison with representative methods from each category: the traditional method PocketOptimizer [20], and three recent deep learning-based models FAIR [46], RFdiffusion All-Atom [14], and PocketGen [47]. More details on running these methods are in Appendix C

Table 1: Evaluation on CrossDocked2020 test set. Here, *reference* represents the native pocket structure and sequence in the dataset. *SR* denotes the sidechain relaxation for the additional post-processing. We highlight the best two results with **bold text** and underlined text, respectively. The scRMSD results for PocketOpt are omitted, as the method keeps the protein backbone structures fixed. $*$ indicates that *SR* only affects sidechain conformations.

Methods	Binding Energy Vina Score (↓)	Binding Energy GlideSP (↓)	Structure Validity scRMSD (↓)	Sequence Plausibility AAR (↑)	Evolutionary Plausibility ESM-2 ppl (↓)
reference	-7.17	-6.60	0.65	-	2.36
PocketOpt	-6.86	-6.33	-	25.86%	2.95
FAIR	-7.01	-6.45	0.85	14.55%	3.34
RFdiffusionAA	-6.93	-6.42	1.16	**37.62%**	**2.49**
PocketGen	-7.19	-6.64	0.79	14.83%	3.31
PocketGen *w/o SR*	-6.48	-6.19	*	*	*
PocketX-Base	-7.21	-6.69	**0.67**	28.9%	2.87
PocketX-DPO	-7.87	-7.16	0.80	27.21%	2.93
PocketX-GRPO	**-8.43**	**-7.85**	0.68	34.89%	2.56

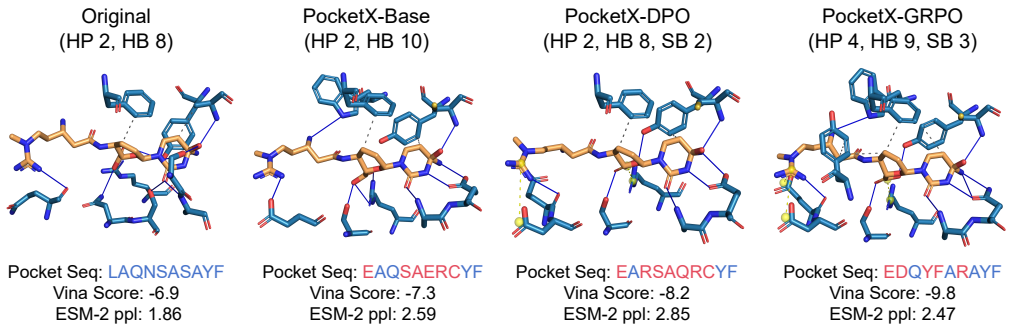

Figure 2: Visualization of protein–ligand interaction analysis for small molecule (PDB ID: 1WN6). HP indicates hydrophobic interactions, HB signifies hydrogen bonds, and SB denotes salt bridges, which are depicted by gray dashed lines, blue solid lines, and yellow dashed lines, respectively. Pocket sequence is shown, with red residues indicating sites that diverge from the native sequence.

Experimental Results As shown in Table 1, Fig. 3, and Fig. 4, PocketX-GRPO achieves the lowest binding energies while maintaining competitive structural validity and evolutionary plausibility. These consistent improvements over PocketX-Base and PocketX-DPO demonstrate that GRPO is an effective optimization paradigm, steering generation under biophysical constraints.

Interaction Analysis As shown in Fig. 2, we use PLIP [30] to analyze the pocket–ligand interactions and compare them with the native pattern. Across different variants, PocketX introduces additional physically plausible contacts; notably, GRPO fine-tuning (vs. DPO) enriches hydrophobic and electrostatic interactions—e.g., via Phenylalanine and Aspartic acid—leading to more favorable binding energy and evolutionary plausibility, demonstrating the advantage of GRPO.

4 Conclusions

We propose PocketX, a diffusion-based generative model that integrates online reinforcement learning for protein pocket design. Using Group Relative Policy Optimization (GRPO), PocketX consistently steers generation toward pockets that satisfy the desired biophysical properties, as confirmed by our experiments. The framework is modular and readily accommodates additional reward signals. For example, predicted confidence scores from AlphaFold3 [1] and affinity estimates from Boltz-2 [22] could be incorporated in an online RL setting as the guidance of protein-ligand complex structures and binding affinity. Exploring these extensions is a central focus of our future work and is expected to enhance PocketX's performance even further.

Acknowledgments and Disclosure of Funding

We acknowledge the financial support from the National Key R&D program of China (grant no. 2023YFF1205103) and the National Natural Science Foundation of China (grant no. 32370657).

References

[1] 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*, 630(8016):493–500, 2024.

[2] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.

[3] Yaoxi Chen, Quan Chen, and Haiyan Liu. Depact and pacmatch: A workflow of designing de novo protein pockets to bind small molecules. *Journal of Chemical Information and Modeling*, 62(4):971–985, 2022.

[4] Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022.

[5] Justas Dauparas, Gyu Rie Lee, Robert Pecoraro, Linna An, Ivan Anishchenko, Cameron Glasscock, and David Baker. Atomic context-conditioned protein sequence design using ligandmpnn. *Nature Methods*, pages 1–7, 2025.

[6] Justas Dauparas, Gyu Rie Lee, Robert Pecoraro, Linna An, Ivan Anishchenko, Cameron Glasscock, and David Baker. Atomic context-conditioned protein sequence design using ligandmpnn. *Nature Methods*, pages 1–7, 2025.

[7] Jiayi Dou, Anastassia A Vorobieva, William Sheffler, Lindsey A Doyle, Hahnbeom Park, Matthew J Bick, Binchen Mao, Glenna W Foight, Min Yen Lee, Lauren A Gagnon, et al. De novo design of a fluorescence-activating β-barrel. *Nature*, 561(7724):485–491, 2018.

[8] 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.

[9] Paul G Francoeur, Tomohide Masuda, Jocelyn Sunseri, Andrew Jia, Richard B Iovanisci, Ian Snyder, and David R Koes. Three-dimensional convolutional neural networks and a cross-docked data set for structure-based drug design. *Journal of chemical information and modeling*, 60(9):4200–4215, 2020.

[10] Siyi Gu, Minkai Xu, Alexander Powers, Weili Nie, Tomas Geffner, Karsten Kreis, Jure Leskovec, Arash Vahdat, and Stefano Ermon. Aligning target-aware molecule diffusion models with exact energy optimization. *Advances in Neural Information Processing Systems*, 37:44040–44063, 2024.

[11] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

[12] Brian L Hie, Varun R Shanker, Duo Xu, Theodora UJ Bruun, Payton A Weidenbacher, Shaogeng Tang, Wesley Wu, John E Pak, and Peter S Kim. Efficient evolution of human antibodies from general protein language models. *Nature biotechnology*, 42(2):275–283, 2024.

[13] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

[14] 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.

[15] 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.

[16] Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *arXiv preprint arXiv:2505.05470*, 2025.

[17] Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5(5):5, 2017.

[18] 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.

[19] Gilles Marcou and Didier Rognan. Optimizing fragment and scaffold docking by use of molecular interaction fingerprints. *Journal of chemical information and modeling*, 47(1):195–207, 2007.

[20] Jakob Noske, Josef Paul Kynast, Dominik Lemm, Steffen Schmidt, and Birte Höcker. Pocketoptimizer 2.0: A modular framework for computer-aided ligand-binding design. *Protein Science*, 32(1):e4516, 2023.

[21] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[22] Saro Passaro, Gabriele Corso, Jeremy Wohlwend, Mateo Reveiz, Stephan Thaler, Vignesh Ram Somnath, Noah Getz, Tally Portnoi, Julien Roy, Hannes Stark, et al. Boltz-2: Towards accurate and efficient binding affinity prediction. *BioRxiv*, pages 2025–06, 2025.

[23] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[24] Dingming Peng, Na Li, Wenting He, Kim Ryun Drasbek, Tao Xu, Mingshu Zhang, and Pingyong Xu. Improved fluorescent proteins for dual-colour post-embedding clem. *Cells*, 11(7):1077, 2022.

[25] 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.

[26] Alfredo Quijano-Rubio, Hsien-Wei Yeh, Jooyoung Park, Hansol Lee, Robert A Langan, Scott E Boyken, Marc J Lajoie, Longxing Cao, Cameron M Chow, Marcos C Miranda, et al. De novo design of modular and tunable protein biosensors. *Nature*, 591(7850):482–487, 2021.

[27] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 3505–3506, 2020.

[28] Milong Ren, ZaiKai He, and Haicang Zhang. Multi-objective antibody design with constrained preference optimization. In *The Thirteenth International Conference on Learning Representations*, 2025.

[29] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[30] Philipp Schake, Sarah Naomi Bolz, Katja Linnemann, and Michael Schroeder. Plip 2025: introducing protein–protein interactions to the protein–ligand interaction profiler. *Nucleic Acids Research*, 53(W1):W463–W465, 05 2025.

[31] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[32] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[33] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

[34] Martin Steinegger and Johannes Söding. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028, 2017.

[35] Andre C Stiel, Mehdi Nellen, and Birte Höcker. Pocketoptimizer and the design of ligand binding sites. In *Computational Design of Ligand Binding Proteins*, pages 63–75. Springer, 2016.

[36] AtomX Team and Haicang Zhang. Atomx: An end-to-end generative model for full-atom protein design. 2025.

[37] Christine E Tinberg, Sagar D Khare, Jiayi Dou, Lindsey Doyle, Jorgen W Nelson, Alberto Schena, Wojciech Jankowski, Charalampos G Kalodimos, Kai Johnsson, Barry L Stoddard, et al. Computational design of ligand-binding proteins with high affinity and selectivity. *Nature*, 501(7466):212–216, 2013.

[38] 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.

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[40] Xinyou Wang, Zaixiang Zheng, Fei Ye, Dongyu Xue, Shujian Huang, and Quanquan Gu. Diffusion language models are versatile protein learners. In *International Conference on Machine Learning*, pages 52309–52333. PMLR, 2024.

[41] Xinyou Wang, Zaixiang Zheng, Fei YE, Dongyu Xue, Shujian Huang, and Quanquan Gu. DPLM-2: A multimodal diffusion protein language model. In *The Thirteenth International Conference on Learning Representations*, 2025.

[42] 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, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, 2023.

[43] Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, et al. Dancegrpo: Unleashing grpo on visual generation. *arXiv preprint arXiv:2505.07818*, 2025.

[44] Bowen Yu, Jiao Liu, Zhanyuan Cui, Chu Wang, Peipei Chen, Chentong Wang, Yanzhe Zhang, Xingxing Zhu, Ze Zhang, Shichao Li, et al. De novo design of light-responsive protein–protein interactions enables reversible formation of protein assemblies. *Nature Chemistry*, pages 1–10, 2025.

[45] Alexandre Zanghellini, Lin Jiang, Andrew M Wollacott, Gong Cheng, Jens Meiler, Eric A Althoff, Daniela Röthlisberger, and David Baker. New algorithms and an in silico benchmark for computational enzyme design. *Protein Science*, 15(12):2785–2794, 2006.

[46] Zaixi Zhang, Zepu Lu, Hao Zhongkai, Marinka Zitnik, and Qi Liu. Full-atom protein pocket design via iterative refinement. *Advances in Neural Information Processing Systems*, 36:16816–16836, 2023.

[47] Zaixi Zhang, Wan Xiang Shen, Qi Liu, and Marinka Zitnik. Efficient generation of protein pockets with pocketgen. *Nature Machine Intelligence*, 6(11):1382–1395, 2024.

[48] Lin Zheng, Jianbo Yuan, Lei Yu, and Lingpeng Kong. A reparameterized discrete diffusion model for text generation. *arXiv preprint arXiv:2302.05737*, 2023.

[49] Xiangxin Zhou, Dongyu Xue, Ruizhe Chen, Zaixiang Zheng, Liang Wang, and Quanquan Gu. Antigen-specific antibody design via direct energy-based preference optimization. *Advances in Neural Information Processing Systems*, 37:120861–120891, 2024.

[50] Tian Zhu, Milong Ren, and Haicang Zhang. Antibody design using a score-based diffusion model guided by evolutionary, physical and geometric constraints. In *Forty-first International Conference on Machine Learning*, 2024.

A Details of Diffusion Processes

For the completeness of our study, we provide a brief introduction to diffusion and model architectures of the denoiser.

A.1 Diffusion Process for Sequence

For the protein sequence, we employ *absorbing* discrete diffusion framework [2, 48], a widely used diffusion model in protein design [40, 41]. Let $\mathrm{Cat}(\mathbf{v})$ be a categorical distribution on sequence $\mathbf{v}$. The forward process of discrete diffusion defines a Markov process governed by the transition kernel

$$q(\mathbf{v}^{(t)} \mid \mathbf{v}^{(t-1)}) = \mathrm{Cat}(\mathbf{v}^{(t)}; \beta_t \mathbf{v}^{(t-1)} + (1-\beta_t)\mathbf{q}_{\text{noise}}), \tag{7}$$

where $0 \ll \beta_t < 1$ is the noise schedule controlling the degree of corruption at timestep t, and it gradually perturbs the data $\mathbf{v}^{(0)} \sim q(\mathbf{v}^{(0)})$ into a stationary distribution $\mathbf{v}^{(\mathrm{T})} \sim \mathbf{q}_{\text{noise}}$. For absorbing diffusion, $\mathbf{q}_{\text{noise}}$ is the point mass with all of the probability on the mask state.

A.2 Diffusion Process for Structure

For the protein structure, we employ the EDM [13] framework to model the diffusion process. Let us denote the protein structure distribution by $q(\mathbf{x}^{(0)})$, with standard deviation $\sigma_{\text{struc\_data}}$, and consider the family of mollified distributions $q(\mathbf{x}^{(t)}; \sigma_t)$ obtained by the forward process

$$q(\mathbf{x}^{(t)} \mid \mathbf{x}^{(t-1)}) = \mathcal{N}(\mathbf{x}^{(t)}; \mathbf{x}^{(t-1)}, (\sigma_t^2 - \sigma_{t-1}^2)\mathbf{I}), \tag{8}$$

where $\mathcal{N}$ is Gaussian distribution. For $\sigma_t \gg \sigma_{\text{struc\_data}}$, $q(\mathbf{x}^{(t)}; \sigma_t)$ is practically indistinguishable from pure Gaussian noise.

A.3 Details of Model Architectures

While previous protein generators typically use small equivariant neural networks, we take inspiration from AlphaFold3 [1] and utilize a non-equivariant network. Our model comprises two primary components: the InputEmbedder module and the Diffusion module. The Diffusion module consists of an AtomTransformer encoder, a Diffusion transformer, and an AtomTransformer decoder. While all three components are constructed as transformer stacks, the AtomTransformer encoder and decoder operate on atom-level representations, whereas the DiT [23] operates on token-level representations. At each time step t, the network receives the noisy sequence $\mathbf{v}^{(t)}$ and structure $\mathbf{x}^{(t)}$ as inputs, and predicts the denoised sequence $\mathbf{v}^{(0)}$ and structure $\mathbf{x}^{(0)}$.

B Details of Implementation

B.1 Pretraining Details

For PocketX-Base model training, we adopted the DeepSpeed [27] strategy and used the Adam optimizer [17] with a learning rate of 0.0001 and parameters β values of $(0.9, 0.999)$. The pretraining was conducted on 4 NVIDIA 80G H100 GPUs with a batch size of 64 and a gradient norm clipping value of 1.0, and achieved convergence within 100k steps.

B.2 Reinforcement Learning Details

For DPO and GRPO reinforcement learning, the pretrained base model was optimized using the Adam optimizer with an initial learning rate of 5×10^{-5} and $\beta = (0.95, 0.999)$. All other hyperparameter settings were kept consistent with those used in pretraining. Following previous works [10, 49], we construct preference-optimized datasets for pocket and ligand complexes. We generated 128 pockets for each ligand in the fine-tuning dataset, and the reward metrics, such as binding energy and evolutionary plausibility, are calculated for these pockets. The highest-scoring samples under the reward function were treated as preferred, in contrast to the lowest-scoring samples, which were treated as dispreferred. Training of the PocketX-DPO model was conducted on 4 NVIDIA H100 GPUs, achieving convergence in 30k steps. We configured the group size as $G = 8$ for

the reinforcement learning process of GRPO, generating eight candidate protein pockets for each ligand molecule. This setting offered a reliable optimization signal while maintaining acceptable computational overhead. The KL ratio β in the GRPO loss function is set as 0.05. The PocketX-GRPO model was trained on 4 NVIDIA H100 GPUs and converged within 80k steps.

C Details of Baseline Methods

PocketOptimizer [20] is a physics-based computational approach for protein design, which predicts mutations within protein binding pockets to enhance affinity for a target ligand. In this study, we employ its most recent release, PocketOptimizer 2.0. The typical workflow of PocketOptimizer involves four key stages: preparing the protein structure, sampling conformational flexibility, calculating energetic contributions, and generating candidate design solutions. During the energy evaluation step, both packing energies and ligand-binding energies are taken into account. In line with the original protocol, the protein backbone was held fixed throughout the design process. We use the test script provided in the GitHub repository (`https://github.com/Hoecker-Lab/pocketoptimizer`). All the hyperparameters we used are default.

RFdiffusion All-Atom [14] is the latest iteration of RFDiffusion, combining residue-level representations for amino acids with atomic representations for other molecular groups. It supports modeling diverse molecular complexes, such as proteins with small molecules, metals, nucleic acids, or covalent modifications. Starting from the random noise of residues around target molecules, it can directly generate the binding protein backbone. However, to complete protein design, especially pocket design, explicit residue identities are required. Thus, LigandMPNN [6], the recently updated version of ProteinMPNN [4], is employed to predict the amino acid type at each position. We use the pretrained model and test script provided in the GitHub repository (`https://github.com/baker-laboratory/rf_diffusion_all_atom`). All the hyperparameters are default.

FAIR [46] is an algorithm for the co-design of pocket sequences and structures. It performs a two-stage process in a coarse-to-fine fashion, starting with backbone refinement and proceeding to full-atom refinement, including side chains, to generate full-atom pockets. We employed FAIR from its GitHub repository (`https://github.com/zaixizhang/FAIR`) with all default hyperparameters.

PocketGen [47] is a deep generative method developed for efficient protein pocket design. It employs a co-design approach in which both the sequence and structure of a protein pocket are predicted from the ligand and the surrounding protein scaffold (excluding the pocket itself). The model architecture consists of two main components: a bilevel graph transformer and a sequence refinement module. We use the pretrained model checkpoint provided in the GitHub repository (`https://github.com/zaixizhang/PocketGen`). All the hyperparameters we used are default.

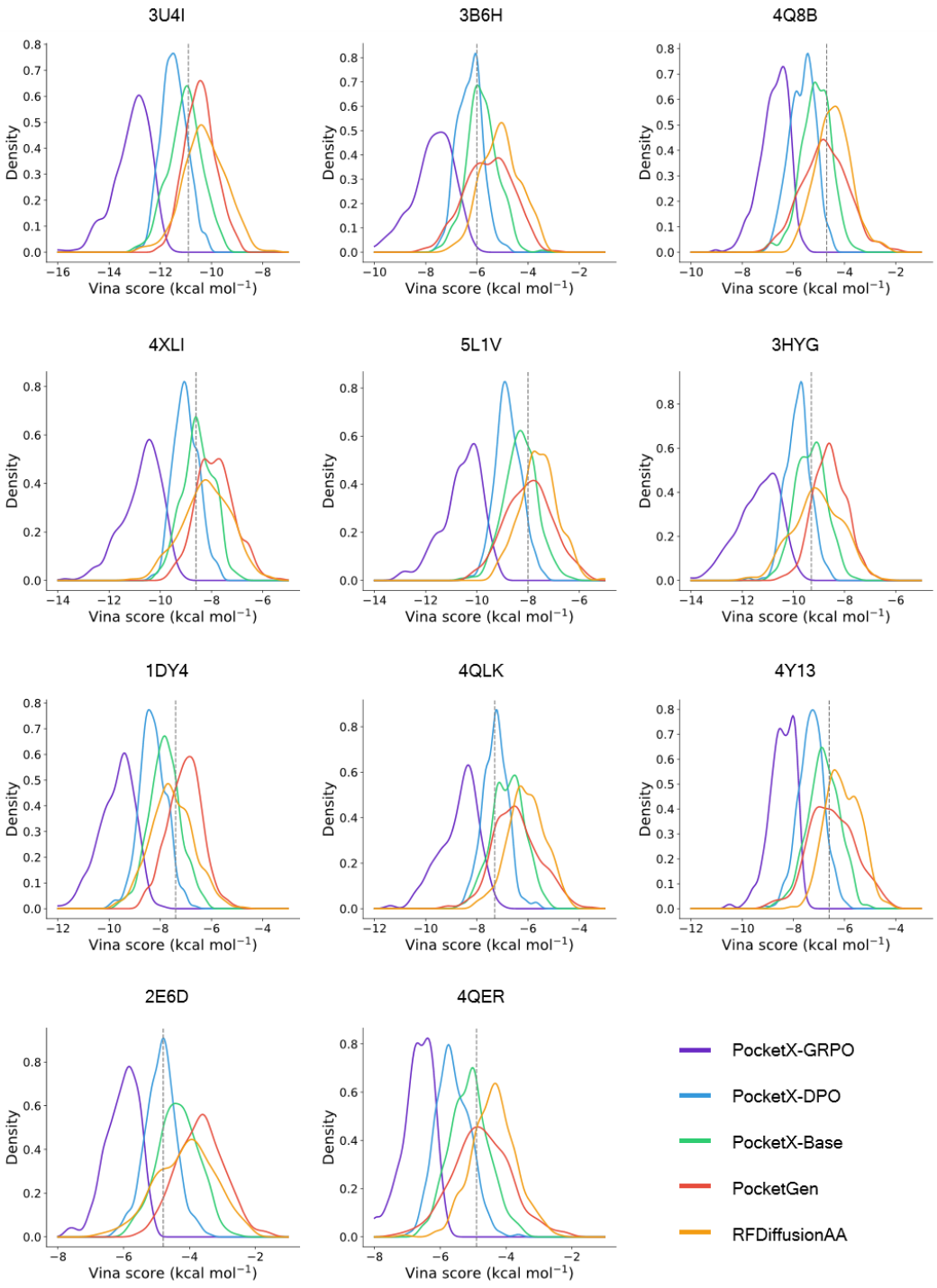

Figure 3: Pocket binding affinity distributions of PocketX and baseline methods for the target molecules in PDB. We mark the Vina Score of the original pocket with the vertical dotted lines. For each method, we sample 100 pockets for each target ligand.

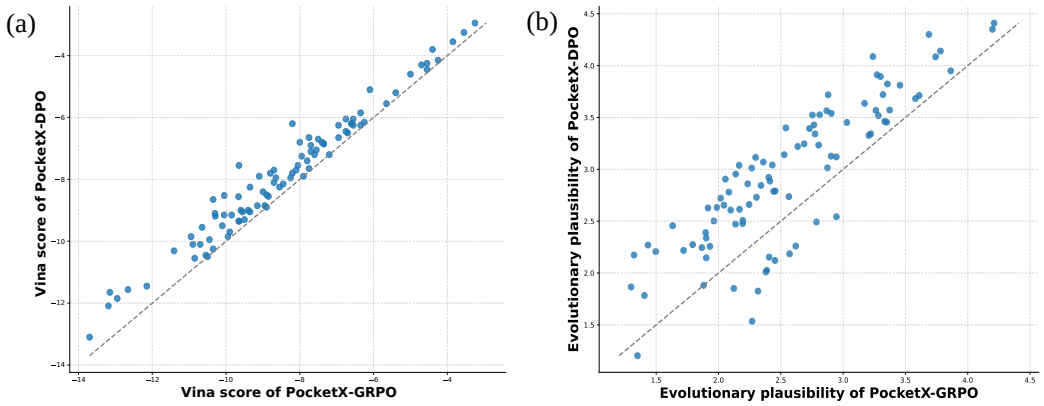

Figure 4: Head-to-head comparison across reward signals. Left: AutoDock Vina score; Right: Evolutionary plausibility.