OMTRA: A Multi-Task Generative Model for Structure-Based Drug Design

Ian Dunn* **Liv Toft†** **Tyler Katz†** **Juhi Gupta†** **Riya Shah*** **Ramith Hettiarachchi†**

David R. Koes*

Abstract

Structure-based drug design (SBDD) focuses on designing small-molecule ligands that bind to specific protein pockets. Computational methods are integral in modern SBDD workflows and often make use of virtual screening methods via docking or pharmacophore search. Modern generative modeling approaches have focused on improving novel ligand discovery by enabling *de novo* design. In this work, we recognize that these tasks share a common structure and can therefore be represented as different instantiations of a consistent generative modeling framework. We propose a unified approach in OMTRA, a multi-modal flow matching model that flexibly performs many tasks relevant to SBDD, including some with no analogue in conventional workflows. Additionally, we curate a dataset of 500M 3D molecular conformers, complementing protein–ligand data and expanding the chemical diversity available for training. OMTRA obtains state-of-the-art performance on pocket-conditioned *de novo* design and docking; however, the effects of large-scale pretraining and multi-task training are modest. All code, trained models, and dataset for reproducing this work are available at `https://github.com/gnina/OMTRA`

1 Introduction

Deep generative models have rapidly advanced our ability to design and study molecules; this progress will have broad applications. By learning distributions over chemical structures, these models enable tasks ranging from probing biological mechanisms to accelerating therapeutic and general chemical discovery [1–4].

We focus here on structure-based drug design (SBDD), where the goal is to generate small molecules that bind to a given protein pocket of known structure. Generative approaches have been impactful in two main directions: structure prediction and *de novo* design. In structure prediction, models propose ligand binding poses [5, 6] or protein–ligand complexes [7–10], improving over classical docking methods by directly sampling plausible structures instead of relying on slow search algorithms. In *de novo* design, generative models sample the identity of the molecules themselves, sometimes in addition to the 3D structure. De novo generative models can also be conditioned on desired properties such as a binding partner [11–18].

Current state of the art methods for both structure prediction and *de novo* design are primarily *transport-based generative models*—a broad class that includes diffusion [19–21], flow matching [22–24], and stochastic interpolant models [25]. These methods also primarily use attributed point-cloud or sequence-based representations of molecules. These approaches treat molecular systems as objects composed of distinct modalities, and may therefore be additionally characterized as *multi-modal transport-based models*. Atoms have both continuous positions and discrete identities [12, 15, 26, 27]. Protein residues are often described similarly but their structure is sometimes decomposed into a product space of rotations, translations, and/or torsional angles [28–31]. These methods

*Department of Computational and Systems Biology, School of Medicine, University of Pittsburgh

†Ray and Stephanie Lane Computational Biology Department, School of Computer Science, Carnegie Mellon University

Machine Learning for Structural Biology Workshop

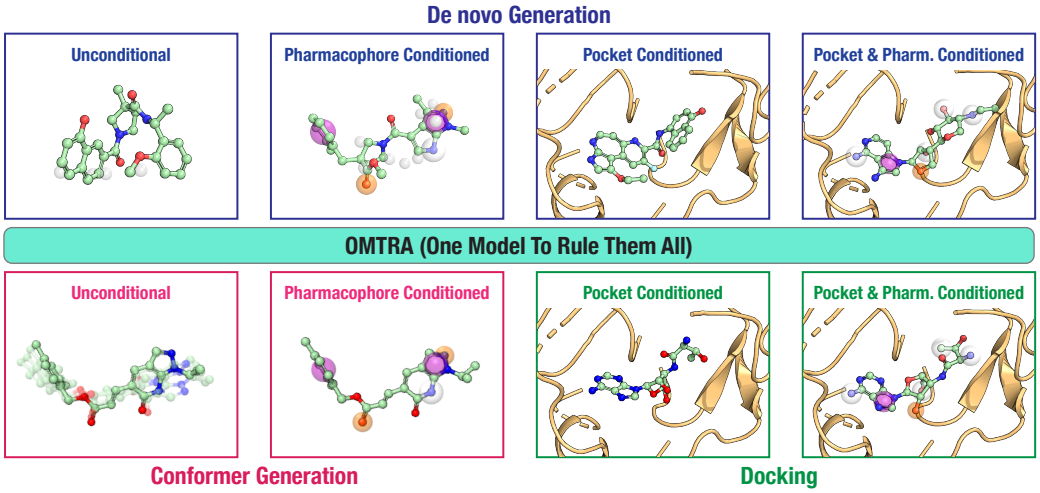

Figure 1: **OMTRA: A Flexible Multi-Task Generative Model for Structure-Based Drug Design.** OMTRA is capable of performing *de novo* design, conformer generation and docking. It can be guided by conditioning on structural information, such as a protein pocket and pharmacophores.

simultaneously apply transport-based modeling to each modality. To our knowledge, this unifying perspective has not been made explicit. After recognizing that many applications or generative tasks may be formulated under one framework, a natural question therefore arises: could one model be trained for many tasks, and could this yield improved performance from transfer learning?

Building on this idea, we introduce **OMTRA**, a multi-task generative model for SBDD. OMTRA supports ligands, protein pockets, and pharmacophores, co-factors, ions, and post-translation modifications. Tasks can be defined by dividing modalities into those generated, held fixed as conditioning information, and those absent. OMTRA is capable of handling arbitrary modality partitions and therefore can perform molecular docking, de novo design, conformer generation, pharmacophore-conditioned design, and related tasks. OMTRA's architecture enables parameter sharing across tasks and learning from disparate data sources.

Finally, we contribute a large-scale dataset of 500M 3D molecular conformers, constructed from public chemical libraries and released in a deep learning–ready format. This dataset expands chemical diversity available for training and complements protein–ligand data, facilitating multi-task learning across heterogeneous molecular modalities.

2 Methods

2.1 Flow Matching

Flow matching [23, 25, 22, 24] prescribes a method to interpolate between two probability distributions. This is accomplished modeling a family of intermediate distributions $\{p_t\}_{t\in[0,1]}$ with $p_0 = q_{\text{source}}$ and $p_1 = q_{\text{target}}$. The general strategy is to draw samples from a tractable q_{source} and use a learned process p_t to transport them to samples from an otherwise intractable q_{target}. For continuous variables $x \in \mathbb{R}^d$, the marginal process is sampled by a vector field defining an ordinary differential equation $u_t(x) = \frac{dx}{dt}$. For discrete data $x \in \mathcal{V}^N$ (sequences of tokens from a vocabulary $\mathcal{V}$), trajectories are governed by continuous-time Markov chains (CTMCs) [13, 32]. In both cases the sampling process can be parameterized by a neural network trained to approximate the final system state given the current state $\mathbb{E}[x_1|x_t]$.

Multi-modal flow matching is a powerful extension of this framework wherein complex objects are modeled as collections of distinct dependent variables, referred to as modalities. For example, atom positions and atom types form two modalities for which joint sampling can enable *de novo* molecule generation. Multi-modal flow matching [12, 13] extends the base flow matching framework to sample such data structures. One neural network is trained with multiple prediction heads to minimize a weighted sum of flow matching losses on each modality. Modalities in the system are generated by simultaneous sampling of an ODE or CTMC on each modality. A full description of our flow matching formulation is provided in Appendix A7.

2.2 Problem Setup

We represent a biomolecular system cropped to 8Å around a ground-truth ligand as a heterogeneous graph $G = (V, E)$ with node types $\mathcal{T}_V = \{\text{ligand atom}, \text{protein atom}, \text{pharmacophore}, \text{other}\}$ and edge types $\mathcal{T}_E$ (e.g., ligand–ligand, ligand–protein). Each node or edge carries one or more *modalities*, discrete (e.g., atom type, bond order) or continuous (e.g., 3D coordinates), so that $\mathcal{M} = \mathcal{M}_{\text{disc}} \cup \mathcal{M}_{\text{cont}}$. A *task* τ specifies a partition of modalities $\mathcal{M} = \mathcal{M}^{\tau}_{\text{gen}} \cup \mathcal{M}^{\tau}_{\text{cond}} \cup \mathcal{M}^{\tau}_{\text{abs}}$ referred to as generated, conditioning, and absent modalities, respectively. Generated modalities, $\mathcal{M}^{\tau}_{\text{gen}}$ are sampled by a flow matching model.

An instantiation of OMTRA requires choosing a set of tasks $\mathcal{T} = \{\tau_1, \ldots, \tau_K\}$ to support. If OMTRA is instantiated as a multi-task model (i.e., $|\mathcal{T}| > 1$), weights are shared across tasks. See Appendix A4 for a full formalization.

A full list of modalities and tasks currently supported, as well as tasks we plan to support in the near feature, are provided in Appendix A8.

2.3 Architecture

OMTRA extends FlowMol3 [12], a geometric graph neural network, to heterogeneous graphs with type-specific convolutions. Nodes $i \in V$ carry Cartesian positions $x_i \in \mathbb{R}^3$, scalar features $s_i \in \mathbb{R}^{d_s}$, and vector features $v_i \in \mathbb{R}^{d_v \times 3}$. Edges $(i, j) \in E$ are typed; ligand–ligand and other–other edges additionally carry scalar features $e_{ij} \in \mathbb{R}^{d_e}$. Operations on (x_i, v_i) are SE(3)-equivariant, while operations on (s_i, e_{ij}) are SE(3)-invariant. Node vectors v_i are first-order geometric vectors relative to x_i. Equivariance is enabled by a variant of GVPs [33] introduced in Dunn and Koes [12].

A *graph convolution* consists of edge-type–specific message functions ϕ_r that generate messages $m^r_{i \leftarrow j}$ based on $(s_i, s_j, v_i, v_j, e_{ij}, x_i - x_j)$, followed by node-type–specific update functions ψ_α that integrate the aggregated messages M_i to update (s_i, v_i). After 2 graph convolutions, positions and edge features are updated via node-wise GVPs and edge-wise MLPs. OMTRA repeats this sequence (called a *ConvolutionBlock*) 4 times. For modalities defining positions, the final x_i are taken directly from the last *ConvolutionBlock*. Latent node and edge features are mapped to categorical logits by shallow MLP heads. The architecture is described fully in Appendix A9.

3 Experiments

3.1 Datasets

To enable pretraining on protein-free tasks, we curated the Pharmit Dataset containing 500M 3D ligands and their pharmacophores. The Pharmit Dataset is further described in Appendix A6.1.

For ablation studies we train on protein-ligand complexes from the Plinder dataset [34] using their rigorously designed splits. For fair evaluation of pocket-conditioned *de novo* design, we also train and evaluate OMTRA on the Crossdocked dataset [35], using the train/test splits from Luo et al. [36]. These splits are less rigorous than those of the original Crossdocked authors and likely contain information leakage, but they are widely reused and thus allow direct comparison to prior work. For comparing OMTRA to other docking methods, we use the PoseBusters Benchmark set [37]. This is a moderately difficult docking benchmark comprising 428 diverse protein-ligand complexes. All datasets are described in detail in Appendix A6. We did not take care to remove overlapping examples from the Plinder training set before evaluating on the Posebusters Benchmark Set, and as such our estimates of docking performance on PoseBusters may be optimistic.

3.2 Evaluation and Baselines

We evaluate OMTRA on pocket-conditioned *de novo* design, docking, and pharmacophore conditioning tasks, comparing against established baselines under a unified protocol.

Pocket-Conditioned De Novo Design Generated ligands are assessed with PoseBusters [37] for chemical/geometric plausibility. The fraction of ligands passing all relevant checks in the PoseBusters suite is reported as %PB-Valid. Strain energy (E_{strain}) and protein-ligand interactions are measured using PoseCheck [38]. Posecheck reports the number of hydrogen bond donors, acceptors, and hydrophobic groups on the ligand that are interacting with the protein, normalized by the number of atoms in the ligand. We report "% Interaction Parity" defined as the fraction of generated ligands that have normalized interaction counts equal to or greater than that of their ground-truth ligand. Additionally we report the fraction of ligands that achieve interaction parity and are PB-valid as "% PB-Valid+IP". We compare OMTRA to DrugFlow [28], DiffSBDD [39], TargetDiff [40], and

Pocket2Mol [41], using 100 test pockets with 100 ligands per pocket. Full experimental details of our evaluation methods are in Appendices A10.1.

Docking Docked poses are evaluated with PoseBusters in docking mode, reporting physical plausibility and RMSD < 2Å to the ground truth, along with Top-N success rates. To perform a Top-N analysis on OMTRA, samples are ranked by the Vina scoring function [42] combined with a penalty for docked poses that violate that don't preserve the chirality of the input ligand. OMTRA is trained on Plinder [34] and evaluated on the PoseBusters Benchmark, sampling 40 poses per receptor. Baseline docking methods include AlphaFold3 [7], SurfDock [43], Gnina [44], AutoDock Vina [42]. Performance is reported as the fraction of systems with at least one PoseBusters-valid pose among the top-N predictions (N=1, 5). Full experimental details of our evaluation methods are in Appendices A10.2.

Pharamcophore Conditioning When pharmacophore centers are provided as conditioning information, we report "% Pharm Matching" which is the fraction of pharmacophore centers in the input conditioning information that are "satisfied" by the generated/docked ligand. A pharmacophore condition is satisfied if the generated ligand has a pharmacophore center of the same type within 1Å of the conditioning pharmacophore. We also report "interaction recovery" which is fraction of ligands that make all of the same types of interactions with the same protein residues as the ground-truth ligand. Pharmacophores are described further in Appendix A5.2.

3.3 Ablations

We train multiple variants on OMTRA to investigate the effects of three features. **Ligand-only pretraining:** Single-task OMTRA models are trained for either *de novo* design or docking, with and without protein-free pretraining on the Pharmit dataset. **Multi-task training:** OMTRA is trained jointly on *de novo* design and docking, initialized from Pharmit-pretrained models, and compared to single-task counterparts. For all ablation studies, we train and evaluate on the Plinder dataset. We attempt to match the amount of compute spent on each task between single- and multi- task models. Single-task models are trained for 200k steps and multi-task models are trained for 400k steps, with an even balance between *de novo* design and docking. **Protein + pharmacophore conditioning:** OMTRA is trained to simultaneously use protein and pharmacophore conditioning. This variant can perform design and docking guided by user-supplied, interpretable priors over protein-ligand interaction.

4 Results

OMTRA Performance on *De Novo* Design Table 1 shows that OMTRA surpasses or matches existing *de novo* generative models across all evaluations performed. By building on FlowMol3 [12], a state-of-the-art model for unconditional *de novo* molecule design, OMTRA is able to obtain the highest physical plausibility of methods evaluated ($\approx 90\%$ PB-Valid). The mean strain energy and Vina scores are practically close to that of the training data. OMTRA produces ligands that are both physically plausible and produce native-like interactions with the protein 37% more frequently than the next best method (26 vs 19 %PB-Valid+IP). Although OMTRA improves upon existing *de novo* generative models, there remain practically significant differences between real and the average generated molecule. Most notably, only 26% of samples are both PB-valid and achieve native interaction parity.

OMTRA achieves SOTA re-docking performance Re-docking results on the PoseBusters Benchmark Set in Table 2 show that OMTRA achieves highly accurate re-docking. After sampling and ranking 40 poses for a pocket, 91% of the top-ranked poses are PB-valid; that is they are within 2Å RMSD of the ground-truth ligand and satisfy a suite of additional physical plausibility checks. OMTRA's top-1 docking accuracy exceeds that of all other models evaluated including AlphaFold3 with pocket residues specified [7]. Additional results (Figure A3) show that OMTRA-docked poses are not PB-valid most often due to unconserved tetrahedral chirality. We alleviate this issue by including a penalty for incorrect chirality in our ranking algorithm as done in Abramson et al. [7]. Methods for enforcing chirality preservation [8, 27, 45] exist and would likely further enhance docking performance.

Effects of Pretraining and Multi-Task Training on Interaction and Plausibility Table 3 suggests that the effects of pretraining and multi-task training are mostly modest and inconsistent across tasks. Adding protein-free pretraining improves physical plausibility and interaction parity (+9 and +2

Table 1: **Pocket-Conditioned De Novo Design**. A Multi-Task OMTRA model is compared to existing generative models on the Crossdocked dataset. Metrics are averaged over 100 pockets, with 100 samples drawn per pocket. PB-Valid+IP is the percent of ligands that are both PB-valid and achieve interaction parity. Evaluation details are provided in Section 3.2

	PB-Valid (%)(↑)	E_{strain} (kcal/mol)(↓)	Vina (kcal/mol)(↓)	Interaction Parity (%) (↑)	PB-Valid+IP (%) (↑)
Dataset	97.7	21.3	-7.1	100	97.7
OMTRA	**89.8**	29.5	**-6.8**	**27**	**26**
Pocket2Mol	86.6	**24.0**	-4.5	22	19
DrugFlow	73.4	72.5	-5.9	21	17
TargetDiff	47.7	446	**-6.8**	20	11
DiffSBDD	38.2	546	-2.8	18	7

Table 2: **Docking Success Rates on the PoseBusters Benchmark Set**. AlphaFold3 [7] and Surfdock [43] results are taken directly from their respective publications. Vina results are taken from Buttenschoen et al. [37]. Additional details in Section 3.2.

	$N = 1$		$N = 5$	
	% ≤ 2Å	%PB-Valid	% ≤ 2Å	%PB-Valid
OMTRA	92	**91**	**97**	**96**
AlphaFold3	**93**	84	-	-
SurfDock	78	40	81	79
Gnina	65	64	82	81
Vina	60	58	-	-

percentage points, respectively) but reduces the PB-validity in docking by 4 percentage points. In contrast, the addition of multi-task training impairs *de novo* design while slightly improving docking performance.

Pharmacophore conditioning enables enhanced interaction design The impact of pharmacophore-conditioning on *de novo* ligand design and docking is quantified in Table 4. When provided with pharmacophore conditioning, OMTRA's ability to design protein-ligand interactions is substantially enhanced, indicated by a 10 percentage point increase in interaction recovery for *de novo* design. Similarly, pharmacophore conditioning significantly enhances the quality of docked poses. Additional experiments (Section A1.5) suggest that for *de novo* design, adding 3 pharmacophores reduces the amount of sampling required by 56%. Users with prior knowledge about their target can obtain more accurate predictions from OMTRA with less sampling.

5 Conclusion

OMTRA is a flexible multi-task generative model for structure-based drug design, achieving state-of-the-art performance in both *de novo* design and molecular docking. It uniquely supports simultaneous pocket and pharmacophore conditioning, enabling the incorporation of user-provided knowledge of protein–ligand interactions to improve accuracy and reduce the amount of sampling necessary.

Table 3: **Effect of Pretraining and Multi-Task Training on *De Novo* Design and Docking**. All models are trained and evaluated on Plinder [34] train/test splits. The top row contains metric values computed on the ground-truth protein-ligand systems where applicable. Docking success rates are computed from the top-5 poses per system. OMTRA variations are described in Section 3.3

		De Novo Design				Docking	
Multitask	Pretrained	% PB-Valid	E_{strain}	% IP	% PB-Valid + IP	% RMSD ≤ 2Å	% PB-Valid
		94.0	38.1	100	94.0		
✗	✗	60.8	91.0	13.0	10.0	96.9	92.7
✗	✓	69.1	65.3	15.0	12.0	92.9	88.8
✓	✓	67.0	71.4	9.0	7.0	94.8	91.7

Table 4: **Effect of Pharmacophore Conditioning on Pocket-Conditioned Design and Docking**. Evaluating one OMTRA model on four tasks: pocket-conditioned *de novo* design and docking, with and without additional pharmacophore conditioning. Evaluation is performed on 100 systems from the Plinder test set having > 20 ligand heavy atoms, drawing 100 samples per system. Docking success rates are for the top-5 ligands in each system as ranked by their Vina score.

		Prot Conditioning	Prot + Pharm Conditioning
denovo design	% PB-Valid	67.0	70.3
	interaction recovery	50.8	60.7
	% Pharm Matches	-	97.2
docking	% RMSD $\leq$ 2Å	94.8	99.0
	% PB-Valid	91.7	92.9
	interaction recovery	80.6	90.7
	% Pharm Matches	-	99.5

Our experiments indicate that the benefits of large-scale, protein-free pretraining and multi-task training are modest and not uniformly positive. OMTRA can be trained on multiple tasks while matching the performance of single-task models trained in isolation. Despite a field-wide shift toward multi-task molecular generative models, empirical evidence for meaningful transfer learning remains limited. In our view, whether molecular generative models can reliably leverage transfer across tasks is still an open question. Future work will use OMTRA as a testbed for investigating architectural choices and generative modeling frameworks that may confer this capability.

The current OMTRA architecture represents a minimal extension of an unconditional molecular generative model to a multi-task, heterogeneous-graph setting, leaving substantial room for improvement. Notably, OMTRA can be extended to generate protein structure as an additional modality. This capability would support tasks involving flexible or unknown protein conformations; use-cases in which no holo structure is available and which represent a more realistic and challenging design scenario than the ones considered in our present evaluations. OMTRA's flexible multi-modal formulation thus makes it not only a versatile tool, but also a platform for advancing molecular generative modeling.

Alongside OMTRA, we release the Pharmit Dataset: one of the largest open-source collections of 3D molecules to date. The Pharmit Dataset is stored in a scalable format with a accessible API for efficient programmatic access so that it can readily incorporated into large-scale machine learning applications. All code, trained models, and dataset for reproducing this work are available at `https://github.com/gnina/OMTRA`.

References

[1] Camille Bilodeau, Wengong Jin, Tommi Jaakkola, Regina Barzilay, and Klavs F. Jensen. Generative models for molecular discovery: Recent advances and challenges. *WIREs Computational Molecular Science*, 12(5):e1608, 2022. ISSN 1759-0884. doi: 10.1002/wcms.1608. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/wcms.1608. _eprint: https://wires.onlinelibrary.wiley.com/doi/pdf/10.1002/wcms.1608.

[2] Yuanqi Du, Arian R. Jamasb, Jeff Guo, Tianfan Fu, Charles Harris, Yingheng Wang, Chenru Duan, Pietro Liò, Philippe Schwaller, and Tom L. Blundell. Machine learning-aided generative molecular design. *Nature Machine Intelligence*, 6(6):589–604, June 2024. ISSN 2522-5839. doi: 10.1038/s42256-024-00843-5. URL https://www.nature.com/articles/s42256-024-00843-5. Publisher: Nature Publishing Group.

[3] Jason Yim, Hannes Stärk, Gabriele Corso, Bowen Jing, Regina Barzilay, and Tommi S. Jaakkola. Diffusion models in protein structure and docking. *WIREs Computational Molecular Science*, 14(2):e1711, 2024. ISSN 1759-0884. doi: 10.1002/wcms.1711. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/wcms.1711. _eprint: https://wires.onlinelibrary.wiley.com/doi/pdf/10.1002/wcms.1711.

[4] Alex Morehead, Lazar Atanackovic, Akshata Hegde, Yanli Wang, Frimpong Boadu, Joel Selvaraj, Alexander Tong, Aditi Krishnapriyan, and Jianlin Cheng. How to go with the flow: flow matching in bioinformatics and computational biology, September 2025. URL https://www.authorea.com/users/637193/articles/1320146-how-to-go-with-the-flow-flow-matching-in-bioinformatics-and-computational-biology. Publication Title: Authorea Published: Authorea preprint.

[5] Alex Morehead and Jianlin Cheng. FlowDock: Geometric Flow Matching for Generative Protein-Ligand Docking and Affinity Prediction, March 2025. URL http://arxiv.org/abs/2412.10966. arXiv:2412.10966 [cs].

[6] Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. DiffDock: Diffusion Steps, Twists, and Turns for Molecular Docking, February 2023. URL http://arxiv.org/abs/2210.01776. arXiv:2210.01776 [q-bio].

[7] Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J. Ballard, Joshua Bambrick, Sebastian W. Bodenstein, David A. Evans, Chia-Chun Hung, Michael O'Neill, David Reiman, Kathryn Tunyasuvunakool, Zachary Wu, Akvilė Žemgulytė, Eirini Arvaniti, Charles Beattie, Ottavia Bertolli, Alex Bridgland, Alexey Cherepanov, Miles Congreve, Alexander I. Cowen-Rivers, Andrew Cowie, Michael Figurnov, Fabian B. Fuchs, Hannah Gladman, Rishub Jain, Yousuf A. Khan, Caroline M. R. Low, Kuba Perlin, Anna Potapenko, Pascal Savy, Sukhdeep Singh, Adrian Stecula, Ashok Thillaisundaram, Catherine Tong, Sergei Yakneen, Ellen D. Zhong, Michal Zielinski, Augustin Žídek, Victor Bapst, Pushmeet Kohli, Max Jaderberg, Demis Hassabis, and John M. Jumper. Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature*, 630(8016):493–500, June 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07487-w. URL https://www.nature.com/articles/s41586-024-07487-w. Publisher: Nature Publishing Group.

[8] Jeremy Wohlwend, Gabriele Corso, Saro Passaro, Noah Getz, Mateo Reveiz, Ken Leidal, Wojtek Swiderski, Liam Atkinson, Tally Portnoi, Itamar Chinn, Jacob Silterra, Tommi Jaakkola, and Regina Barzilay. Boltz-1 Democratizing Biomolecular Interaction Modeling, May 2025. URL https://www.biorxiv.org/content/10.1101/2024.11.19.624167v4. Pages: 2024.11.19.624167 Section: New Results.

[9] Chai Discovery, Jacques Boitreaud, Jack Dent, Matthew McPartlon, Joshua Meier, Vinicius Reis, Alex Rogozhnikov, and Kevin Wu. Chai-1: Decoding the molecular interactions of life, October 2024. URL https://www.biorxiv.org/content/10.1101/2024.10.10.615955v2. Pages: 2024.10.10.615955 Section: New Results.

[10] Zhuoran Qiao, Weili Nie, Arash Vahdat, Thomas F. Miller, and Animashree Anandkumar. State-specific protein–ligand complex structure prediction with a multiscale deep generative

model. *Nature Machine Intelligence*, 6(2):195–208, February 2024. ISSN 2522-5839. doi: 10.1038/s42256-024-00792-z. URL https://www.nature.com/articles/s42256-024-00792-z. Publisher: Nature Publishing Group.

[11] Matthew Ragoza, Tomohide Masuda, and David Ryan Koes. Generating 3d molecules conditional on receptor binding sites with deep generative models. *Chemical science*, 13(9): 2701–2713, 2022.

[12] Ian Dunn and David R. Koes. FlowMol3: Flow Matching for 3D De Novo Small-Molecule Generation, August 2025. URL http://arxiv.org/abs/2508.12629. arXiv:2508.12629 [cs].

[13] Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative Flows on Discrete State-Spaces: Enabling Multimodal Flows with Applications to Protein Co-Design, June 2024. URL http://arxiv.org/abs/2402.04997. arXiv:2402.04997 [cs, q-bio, stat].

[14] Julian Cremer, Ross Irwin, Alessandro Tibo, Jon Paul Janet, Simon Olsson, and Djork-Arné Clevert. FLOWR: Flow Matching for Structure-Aware De Novo, Interaction- and Fragment-Based Ligand Generation, May 2025. URL http://arxiv.org/abs/2504.10564. arXiv:2504.10564 [q-bio] version: 2.

[15] Chaitanya K. Joshi, Xiang Fu, Yi-Lun Liao, Vahe Gharakhanyan, Benjamin Kurt Miller, Anuroop Sriram, and Zachary W. Ulissi. All-atom Diffusion Transformers: Unified generative modelling of molecules and materials, May 2025. URL http://arxiv.org/abs/2503.03965. arXiv:2503.03965 [cs].

[16] Bowen Jing, Anna Sappington, Mihir Bafna, Ravi Shah, Adrina Tang, Rohith Krishna, Adam Klivans, Daniel J. Diaz, and Bonnie Berger. Generating functional and multistate proteins with a multimodal diffusion transformer, September 2025. URL https://www.biorxiv.org/content/10.1101/2025.09.03.672144v2. ISSN: 2692-8205 Pages: 2025.09.03.672144 Section: New Results.

[17] Yiming Qin, Manuel Madeira, Dorina Thanou, and Pascal Frossard. DeFoG: Discrete Flow Matching for Graph Generation, June 2025. URL http://arxiv.org/abs/2410.04263. arXiv:2410.04263 [cs].

[18] Leo Tianlai Chen, Zachary Quinn, Madeleine Dumas, Christina Peng, Lauren Hong, Moises Lopez-Gonzalez, Alexander Mestre, Rio Watson, Sophia Vincoff, Lin Zhao, Jianli Wu, Audrey Stavrand, Mayumi Schaepers-Cheu, Tian Zi Wang, Divya Srijay, Connor Monticello, Pranay Vure, Rishab Pulugurta, Sarah Pertsemlidis, Kseniia Kholina, Shrey Goel, Matthew P. DeLisa, Jen-Tsan Ashley Chi, Ray Truant, Hector C. Aguilar, and Pranam Chatterjee. Target sequence-conditioned design of peptide binders using masked language modeling. *Nature Biotechnology*, pages 1–9, August 2025. ISSN 1546-1696. doi: 10.1038/s41587-025-02761-2. URL https://www.nature.com/articles/s41587-025-02761-2. Publisher: Nature Publishing Group.

[19] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep Unsupervised Learning using Nonequilibrium Thermodynamics, November 2015. URL http://arxiv.org/abs/1503.03585. arXiv:1503.03585 [cond-mat, q-bio, stat].

[20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models, December 2020. URL http://arxiv.org/abs/2006.11239. arXiv:2006.11239 [cs, stat].

[21] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-Based Generative Modeling through Stochastic Differential Equations, February 2021. URL http://arxiv.org/abs/2011.13456. arXiv:2011.13456 [cs, stat].

[22] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow Matching for Generative Modeling, February 2023. URL http://arxiv.org/abs/2210.02747. arXiv:2210.02747 [cs, stat].

[23] Alexander Tong, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Kilian Fatras, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport, July 2023. URL http://arxiv.org/abs/2302.00482. arXiv:2302.00482 [cs].

[24] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow, September 2022. URL http://arxiv.org/abs/2209.03003. arXiv:2209.03003 [cs].

[25] Michael S. Albergo, Nicholas M. Boffi, and Eric Vanden-Eijnden. Stochastic Interpolants: A Unifying Framework for Flows and Diffusions, November 2023. URL http://arxiv.org/abs/2303.08797. arXiv:2303.08797 [cond-mat].

[26] Ross Irwin, Alessandro Tibo, Jon Paul Janet, and Simon Olsson. Efficient 3D Molecular Generation with Flow Matching and Scale Optimal Transport, June 2024. URL http://arxiv.org/abs/2406.07266. arXiv:2406.07266 [cs].

[27] Filipp Nikitin, Ian Dunn, David Ryan Koes, and Olexandr Isayev. GEOM-Drugs Revisited: Toward More Chemically Accurate Benchmarks for 3D Molecule Generation, May 2025. URL http://arxiv.org/abs/2505.00169. arXiv:2505.00169 [cs].

[28] Arne Schneuing, Ilia Igashov, Adrian W. Dobbelstein, Thomas Castiglione, Michael Bronstein, and Bruno Correia. Multi-domain distribution learning for de novo drug design, 2025. URL https://arxiv.org/abs/2508.17815.

[29] Guillaume Huguet, James Vuckovic, Kilian Fatras, Eric Thibodeau-Laufer, Pablo Lemos, Riashat Islam, Cheng-Hao Liu, Jarrid Rector-Brooks, Tara Akhound-Sadegh, Michael Bronstein, Alexander Tong, and Avishek Joey Bose. Sequence-Augmented SE(3)-Flow Matching For Conditional Protein Backbone Generation, May 2024. URL http://arxiv.org/abs/2405.20313. arXiv:2405.20313 [cs, q-bio].

[30] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, August 2021. ISSN 1476-4687. doi: 10.1038/s41586-021-03819-2. URL https://www.nature.com/articles/s41586-021-03819-2. Publisher: Nature Publishing Group.

[31] 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, Shan Tie, Vincent Xue, Sarah C. Cowles, Alan Leung, João V. Rodrigues, Claudio L. Morales-Perez, Alex M. Ayoub, Robin Green, Katherine Puentes, Frank Oplinger, Nishant V. Panwar, Fritz Obermeyer, Adam R. Root, Andrew L. Beam, Frank J. Poelwijk, and Gevorg Grigoryan. Illuminating protein space with a programmable generative model. *Nature*, 623(7989):1070–1078, November 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-06728-8. URL https://www.nature.com/articles/s41586-023-06728-8. Publisher: Nature Publishing Group.

[32] Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky T. Q. Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete Flow Matching, July 2024. URL http://arxiv.org/abs/2407.15595. arXiv:2407.15595 [cs].

[33] Bowen Jing, Stephan Eismann, Pratham N. Soni, and Ron O. Dror. Equivariant Graph Neural Networks for 3D Macromolecular Structure, July 2021. URL http://arxiv.org/abs/2106.03843. arXiv:2106.03843 [cs, q-bio].

[34] Janani Durairaj, Yusuf Adeshina, Zhonglin Cao, Xuejin Zhang, Vladas Oleinikovas, Thomas Duignan, Zachary McClure, Xavier Robin, Gabriel Studer, Daniel Kovtun, Emanuele Rossi, Guoqing Zhou, Srimukh Veccham, Clemens Isert, Yuxing Peng, Prabindh Sundareson, Mehmet

Akdel, Gabriele Corso, Hannes Stärk, Gerardo Tauriello, Zachary Carpenter, Michael Bronstein, Emine Kucukbenli, Torsten Schwede, and Luca Naef. Plinder: The protein-ligand interactions dataset and evaluation resource. *bioRxiv*, 2024. doi: 10.1101/2024.07.17.603955. URL https://www.biorxiv.org/content/early/2024/07/19/2024.07.17.603955.1.

[35] 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, September 2020. ISSN 1549-9596. doi: 10.1021/acs.jcim.0c00411. URL https://doi.org/10.1021/acs.jcim.0c00411. Publisher: American Chemical Society.

[36] Shitong Luo, Jiaqi Guan, Jianzhu Ma, and Jian Peng. A 3D Generative Model for Structure-Based Drug Design, November 2022. URL http://arxiv.org/abs/2203.10446. arXiv:2203.10446 [q-bio].

[37] Martin Buttenschoen, Garrett M. Morris, and Charlotte M. Deane. PoseBusters: AI-based docking methods fail to generate physically valid poses or generalise to novel sequences. *Chemical Science*, 15(9):3130–3139, February 2024. ISSN 2041-6539. doi: 10.1039/D3SC04185A. URL https://pubs.rsc.org/en/content/articlelanding/2024/sc/d3sc04185a. Publisher: The Royal Society of Chemistry.

[38] Charles Harris, Kieran Didi, Arian R. Jamasb, Chaitanya K. Joshi, Simon V. Mathis, Pietro Lio, and Tom Blundell. Benchmarking Generated Poses: How Rational is Structure-based Drug Design with Generative Models?, August 2023. URL http://arxiv.org/abs/2308.07413. arXiv:2308.07413 [q-bio].

[39] Arne Schneuing, Charles Harris, Yuanqi Du, Kieran Didi, Arian Jamasb, Ilia Igashov, Weitao Du, Carla Gomes, Tom Blundell, Pietro Lio, Max Welling, Michael Bronstein, and Bruno Correia. Structure-based drug design with equivariant diffusion models. URL http://arxiv.org/abs/2210.13695.

[40] Jiaqi Guan, Wesley Wei Qian, Xingang Peng, Yufeng Su, Jian Peng, and Jianzhu Ma. 3d equivariant diffusion for target-aware molecule generation and affinity prediction, 2023. URL http://arxiv.org/abs/2303.03543.

[41] Xingang Peng, Shitong Luo, Jiaqi Guan, Qi Xie, Jian Peng, and Jianzhu Ma. Pocket2Mol: Efficient Molecular Sampling Based on 3D Protein Pockets, July 2025. URL http://arxiv.org/abs/2205.07249. arXiv:2205.07249 [cs].

[42] 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, August 2021. ISSN 1549-9596. doi: 10.1021/acs.jcim.1c00203. URL https://doi.org/10.1021/acs.jcim.1c00203. Publisher: American Chemical Society.

[43] Duanhua Cao, Mingan Chen, Runze Zhang, Zhaokun Wang, Manlin Huang, Jie Yu, Xinyu Jiang, Zhehuan Fan, Wei Zhang, Hao Zhou, Xutong Li, Zunyun Fu, Sulin Zhang, and Mingyue Zheng. SurfDock is a surface-informed diffusion generative model for reliable and accurate protein–ligand complex prediction. *Nature Methods*, 22(2):310–322, February 2025. ISSN 1548-7105. doi: 10.1038/s41592-024-02516-y. URL https://www.nature.com/articles/s41592-024-02516-y. Publisher: Nature Publishing Group.

[44] Andrew T. McNutt, Yanjing Li, Rocco Meli, Rishal Aggarwal, and David Ryan Koes. GNINA 1.3: the next increment in molecular docking with deep learning. *Journal of Cheminformatics*, 17(1):28, March 2025. ISSN 1758-2946. doi: 10.1186/s13321-025-00973-x. URL https://doi.org/10.1186/s13321-025-00973-x.

[45] Ivan Anishchenko, Yakov Kipnis, Indrek Kalvet, Guangfeng Zhou, Rohith Krishna, Samuel J. Pellock, Anna Lauko, Gyu Rie Lee, Linna An, Justas Dauparas, Frank DiMaio, and David Baker. Modeling protein-small molecule conformational ensembles with PLACER, September 2025. URL https://www.biorxiv.org/content/10.1101/2024.09.25.614868v2. ISSN: 2692-8205 Pages: 2024.09.25.614868 Section: New Results.

[46] Ilia Igashov, Arne Schneuing, Adrian W. Dobbelstein, Irina Morozova, Rebecca Manuela Neeser, Evgenia Elizarova, Philippe Schwaller, Michael M. Bronstein, and Bruno Correia. Large drug discovery model. In *ICLR 2025 Workshop on Generative and Experimental Perspectives for Biomolecular Design*, 2025. URL https://openreview.net/forum?id=fbL6nHy2xb.

[47] Xingang Peng, Fenglin Guo, Ruihan Guo, Jiayu Sun, Jiaqi Guan, Yinjun Jia, Yan Xu, Yanwen Huang, Muhan Zhang, Jian Peng, Xinquan Wang, Chuanhui Han, Zihua Wang, and Jianzhu Ma. Atom-level generative foundation model for molecular interaction with pockets, August 2025. URL https://www.biorxiv.org/content/10.1101/2024.10.17.618827v2. ISSN: 2692-8205 Pages: 2024.10.17.618827 Section: New Results.

[48] Danny Reidenbach, Filipp Nikitin, Olexandr Isayev, and Saee Paliwal. Applications of Modular Co-Design for De Novo 3D Molecule Generation, May 2025. URL http://arxiv.org/abs/2505.18392. arXiv:2505.18392 [cs].

[49] Julian Cremer, Tuan Le, Frank Noé, Djork-Arné Clevert, and Kristof T. Schütt. PILOT: Equivariant diffusion for pocket conditioned de novo ligand generation with multi-objective guidance via importance sampling, May 2024. URL http://arxiv.org/abs/2405.14925. arXiv:2405.14925 [q-bio] version: 1.

[50] Tuan Le, Julian Cremer, Frank Noé, Djork-Arné Clevert, and Kristof Schütt. Navigating the Design Space of Equivariant Diffusion-Based Generative Models for De Novo 3D Molecule Generation, November 2023. URL http://arxiv.org/abs/2309.17296. arXiv:2309.17296 [cs].

[51] openmm/pdbfixer, September 2025. URL https://github.com/openmm/pdbfixer. original-date: 2013-08-29T22:29:24Z.

[52] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

[53] Jocelyn Sunseri and David Ryan Koes. Pharmit: interactive exploration of chemical space. *Nucleic Acids Research*, 44(W1):W442–W448, July 2016. ISSN 0305-1048. doi: 10.1093/nar/gkw287. URL https://doi.org/10.1093/nar/gkw287.

[54] David Ryan Koes and Carlos J. Camacho. Pharmer: Efficient and Exact Pharmacophore Search. *Journal of Chemical Information and Modeling*, 51(6):1307–1314, June 2011. ISSN 1549-9596. doi: 10.1021/ci200097m. URL https://doi.org/10.1021/ci200097m. Publisher: American Chemical Society.

[55] Shuzhe Wang, Jagna Witek, Gregory A. Landrum, and Sereina Riniker. Improving Conformer Generation for Small Rings and Macrocycles Based on Distance Geometry and Experimental Torsional-Angle Preferences. *Journal of Chemical Information and Modeling*, 60(4):2044–2058, April 2020. ISSN 1549-9596. doi: 10.1021/acs.jcim.0c00025. URL https://doi.org/10.1021/acs.jcim.0c00025. Publisher: American Chemical Society.

[56] A. K. Rappe, C. J. Casewit, K. S. Colwell, W. A. III Goddard, and W. M. Skiff. UFF, a full periodic table force field for molecular mechanics and molecular dynamics simulations. *Journal of the American Chemical Society*, 114(25):10024–10035, December 1992. ISSN 0002-7863. doi: 10.1021/ja00051a040. URL https://doi.org/10.1021/ja00051a040. Publisher: American Chemical Society.

[57] Andrew T McNutt, Fatimah Bisiriyu, Sophia Song, Ananya Vyas, Geoffrey R Hutchison, and David Ryan Koes. Conformer generation for structure-based drug design: How many and how good? *Journal of Chemical Information and Modeling*, 63(21):6598–6607, 2023.

[58] 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, February 2017. ISSN 0001-4842. doi: 10.1021/acs.accounts.6b00491. URL https://doi.org/10.1021/acs.accounts.6b00491. Publisher: American Chemical Society.

[59] Shitong Luo, Jiaqi Guan, Jianzhu Ma, and Jian Peng. A 3d generative model for structure-based drug design, November 2022. URL https://arxiv.org/abs/2203.10446. arXiv:2203.10446 [q-bio].

[60] Cédric Bouysset and Sébastien Fiorucci. ProLIF: a library to encode molecular interactions as fingerprints. *Journal of Cheminformatics*, 13(1):72, September 2021. ISSN 1758-2946. doi: 10.1186/s13321-021-00548-6. URL https://doi.org/10.1186/s13321-021-00548-6.

[61] Kexin Zhang, Yuanyuan Ma, Jiale Yu, Huiting Luo, Jinyu Lin, Yifan Qin, Xiangcheng Li, Qian Jiang, Fang Bai, Jiayi Dou, Jie Zheng, Jingyi Yu, and Liping Sun. PhysDock: A Physics-Guided All-Atom Diffusion Model for Protein-Ligand Complex Prediction, June 2025. URL https://www.biorxiv.org/content/10.1101/2025.04.28.650887v4. Pages: 2025.04.28.650887 Section: New Results.

Appendix for OMTRA

Appendix Contents

A1 Extended Results

A1.1 Performance on No-Protein Tasks

In Table A1 we provide the performance of OMTRA on tasks that do not involve protein structures. % PB-Valid is the fraction of molecules passing all PoseBusters [37] checks. % Valid Bond Lengths indicates molecules with physically reasonable bond distances, as determined by PoseBusters. % Chirality Preserved reflects the fraction of tetrahedral stereocenters retaining their configuration (relevant for conformer generation). % Pharm Match is the fraction of molecules in which all ground-truth pharmacophore centers are matched by at least one generated functional group of the same type within 1 Å.

Overall, unconditional generation achieves near-perfect PB-validity and bond lengths but low stereochemistry retention (17.2%), while pharmacophore conditioning improves chirality preservation (17.2% $\rightarrow$ 32.8%) with a slight decline in structural validity.

Table A1: Selected performance metrics for 500 samples per task.

Task	% PB-Valid	% Valid Bond Lengths	% Chirality Preserved	% Pharm Match
Unconditional De Novo Design	99.4	100.0	–	–
Unconditional Conformer Generation	100.0	100.0	17.2	–
Pharm-Conditioned De Novo Design	97.1	99.6	–	98.7
Pharm-Conditioned Conformer Generation	98.0	98.7	32.8	99.3

A1.2 Trajectories

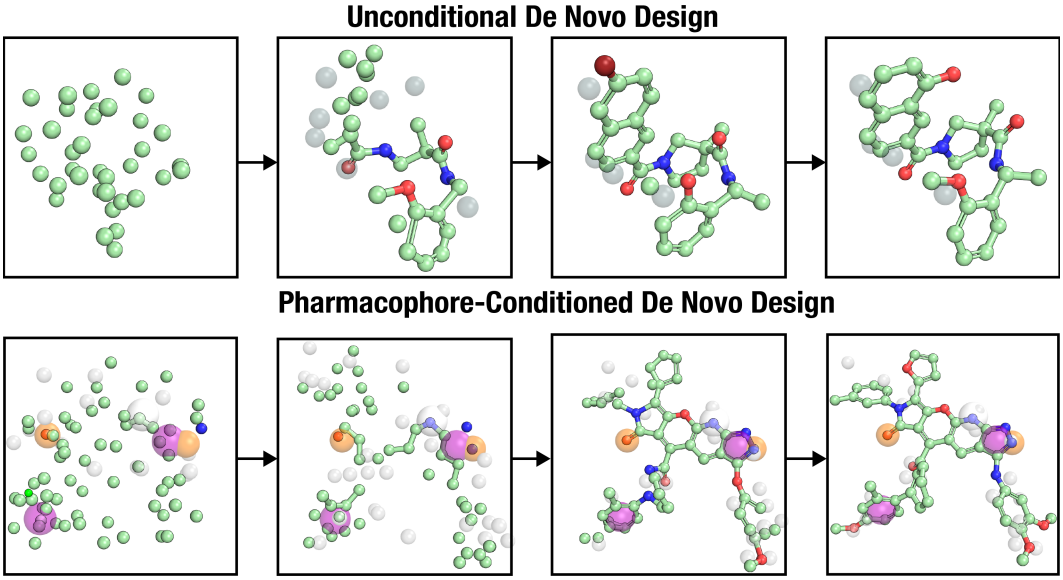

Figure A1: Trajectories of Ligand Generation Using OMTRA. Each row demonstrates the stepwise evolution of ligand structures across OMTRA's sampling trajectory. The top row illustrates unconditional de novo design, where OMTRA generates ligand structures without guiding constraints. The bottom row illustrates pharmacophore-conditioned de novo design, where OMTRA generates ligand structures guided by pre-defined pharmacophores.

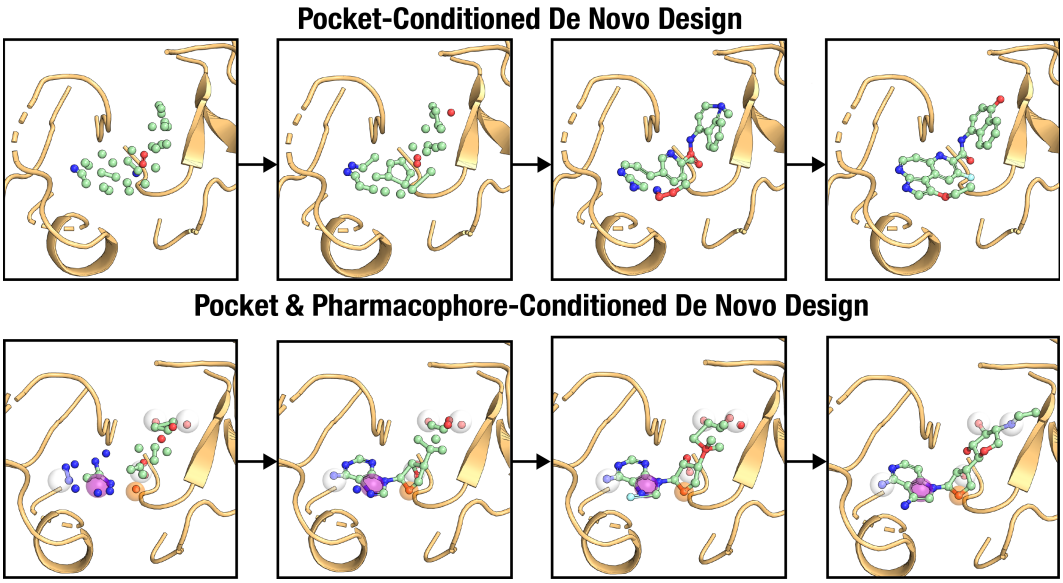

Figure A2: Trajectories of Ligand Generation Inside Pocket Using OMTRA. Each row demonstrates the stepwise evolution of ligand structures across OMTRA's sampling trajectory. The top row illustrates pocket-conditioned de novo design, where OMTRA generates ligand structures guided by the receptor pocket. The bottom row illustrates pocket and pharmacophore-conditioned de novo design, where OMTRA generates ligand structures guided by both the receptor pocket and pre-defined pharmacophores.

A1.3 PoseBusters Checks

We report the pass rates of OMTRA-docked molecules on individual PoseBusters tests in Figure A3. Importantly here we report pass rates on the *average* docked pose. By contrast, in the main paper, we sample poses, rank them, and then compute metrics on the top-N poses for each pocket. The mean value of metrics on docked poses without ranking is substantially worse. Drawing multiple samples and ranking them therefore remains an essential step in obtaining high-quality predictions. This is consistent with other state-of-the-art transport-based generative models.

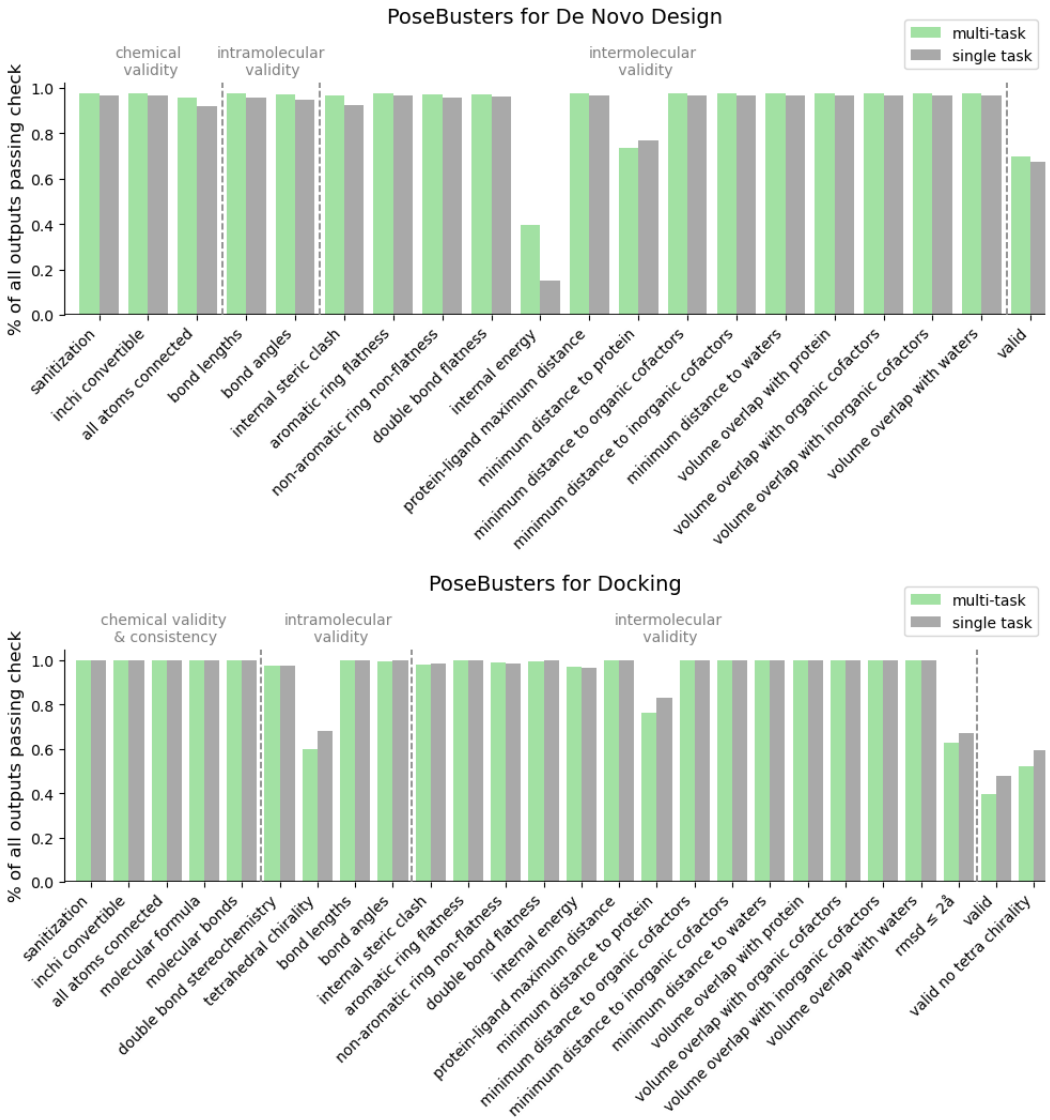

Figure A3: **PoseBusters Checks**. Comparison of multi-task versus single task OMTRA models on PoseBusters checks for *de novo* design (Top) and docking (Bottom). All models were pretrained on unconditional ligand generation tasks. Metrics report the fraction of passing ligands, calculated from 100 samples per pocket across 100 proteins in the Plinder test split.

A1.4 Examples of Top De Novo Ligands

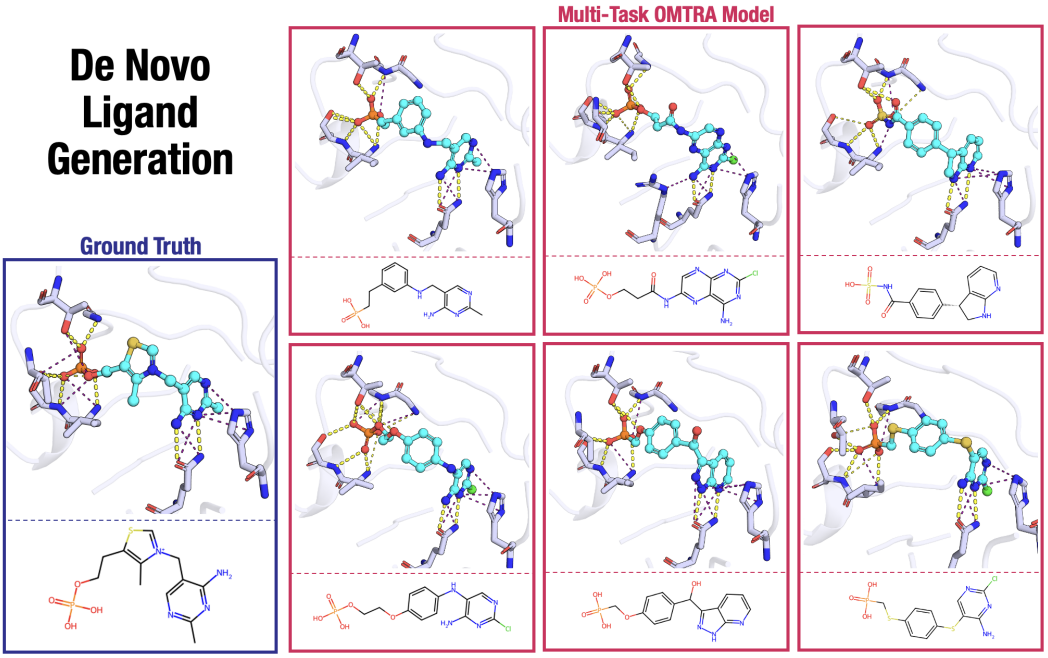

Figure A4: **Top *De Novo* Ligands Generated by OMTRA Multi-Task Model**. Ligand generation was conditioned on the binding pocket of thiamin phosphate synthase (PDB: 1G4S). Protein-ligand contacts are shown as dashed lines. The ground truth ligand, thiamin phosphate (CCD: TPS), is shown in the left panel. *De novo* generated samples (Right) for this target have an interaction recovery rate of 85.7-100%.

A1.5 Effects of Pharmacophore Conditioning

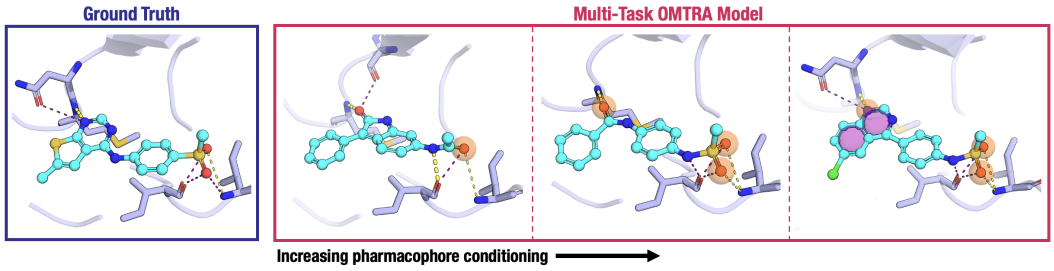

Figure A5: **Pharmacophore-Guided *De Novo* Design**: The left-panel shows a ground-truth complex of a PI5P4K inhibitor (PDB 8BQ4) from the Plinder test set. The three panels on the right side show *de novo* designed ligands that were conditioned on both the pocket structure and varying numbers of pharmacophore centers. Pharmacophore centers are shown as transparent spheres. Polar contacts are shown with dashed lines.

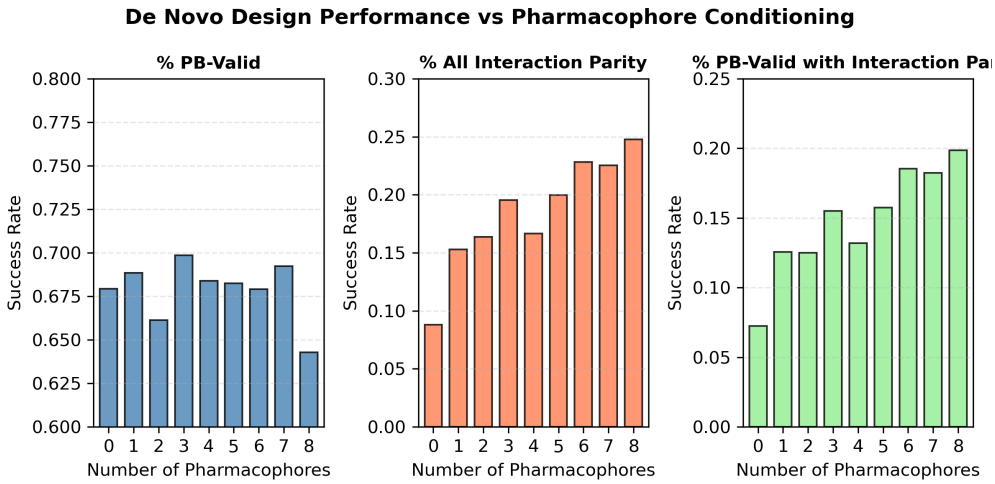

Figure A6: ***De Novo* Design Quality as a Function of Pharmacophore Conditioning**. Each bar plot shows a metric for the quality of designed ligands. The x-axis is the number of pharmacophore centers that were used as conditioning information when designing a ligand. The model used here was trained and evaluated on the Plinder train and test sets, respectively. Approximately 40,000 samples were generated for this figure.

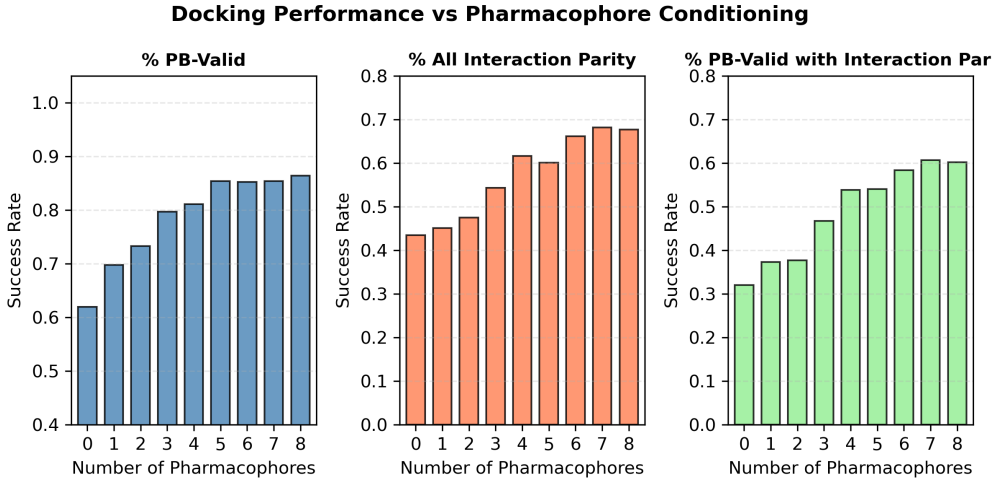

Figure A7: **Docking Success as a Function of Pharmacophore Conditioning**. Each bar plot shows a metric for the quality of docked ligands. The x-axis is the number of pharmacophore centers that were used as conditioning information when docking a ligand. These values are the mean over all samples; no ranking or filtering has been applied to the samples drawn. The model used here was trained and evaluated on the Plinder train and test sets, respectively. Approximately 40,000 samples were generated for this figure.

A2 Related Work

There have been two concurrent works, to our knowledge, that have proposed a variant of multi-task models for structure based drug design. These are the Large Drug Discovery Model (LDDM) [46] and PocketXMol [47]. Both models perform tasks related to small-molecule SBDD. To our knowledge, neither work has presented ablation experiments to measure the effects of multi-task training vs training several single-task models. Additionally, to our knowledge, neither of these works supports pharmacophores as a modality.

Several works have proposed variations of multi-task models in other domains. Campbell et al. [13] proposed a joint protein sequence and backbone model that can perform arbitrary tasks from altering these modalities; forward-folding, inverse-folding, and co-folding. Huguet et al. [29] proposed a

protein structure generative model that could either perform unconditional backbone sampling or sequence-conditioned backbone sampling. Reidenbach et al. [48] trained two variants of the same model for both unconditional conformer generation and unconditional de novo design, but not one model for both tasks.

Cremer et al. [49] and Cremer et al. [14] have proposed unconditional pre-training for pocket-conditioned de novo generative models.

A3 Experiments

Our experiments address three themes: the role of ligand-only pretraining, the impact of multi-task training, and the utility of pharmacophore conditioning.

Ligand-Only Pretraining. We evaluate single-task OMTRA models for pocket-conditioned *de novo* design and molecular docking, each with and without Pharmit pretraining. Pretraining mixes conformer generation (50%) and unconditional design (50%), trained for $\sim$5 days on 2 L40 GPUs. This yields four OMTRA variants.

Multi-Task Training. OMTRA is jointly trained on *de novo* design and docking, initialized from the Pharmit-pretrained models, and compared against Pharmit-pretrained single-task counterparts.

Protein + Pharmacophore Conditioning. We extend OMTRA to simultaneously incorporate protein and pharmacophore context. This enables pocket-conditioned *de novo* design where pharmacophore constraints bias generation toward reproducing specific interactions, and docking where pharmacophore-guided soft constraints influence pose prediction. These tasks, to our knowledge, are unexplored in prior work and demonstrate how OMTRA can guide *de novo* design and docking with interpretable, human-provided priors.

A4 Problem Setup

Graphs. Each biomolecular system is represented as a heterogeneous graph

$$G = (V, E),$$

with node types

$$\mathcal{T}_V = \{\text{ligand atom (L)}, \text{protein atom (P)}, \text{pharmacophore (Ph)}, \text{NPNDE (O)}\},$$

and edge types

$$\mathcal{T}_E = \{\text{ligand–ligand}, \text{ligand–protein}, \text{protein–protein}, \ldots\}.$$

Ligand atoms form a complete subgraph $G[L]$. Explicit hydrogens are omitted unless they are essential for defining the molecule identity as determiend by rdkit; this is described in Section A5.1. NPNDE stands for "Non-Protein Non-Designable Entity"; these are molecules that exist in a system that are not the target ligand nor the protein, but are biologically relevant. These can include ions, co-factors, and post-translational modifications. NPNDEs are described in detail in Section A5.4.

Modalities. A *modality* is any property of the system to be modeled. The modality set is

$$\mathcal{M} = \mathcal{M}_{\text{disc}} \cup \mathcal{M}_{\text{cont}},$$

with $\mathcal{M}_{\text{disc}}$ discrete (e.g., atom types, bond orders) and $\mathcal{M}_{\text{cont}}$ continuous (e.g., 3D coordinates). Each modality $m \in \mathcal{M}$ is attached to a node type $t \in \mathcal{T}_V$ or edge type $t \in \mathcal{T}_E$. Multi-modal flow matching treats these jointly: each $m \in \mathcal{M}$ admits a conditional probability path $\{p_t^m\}_{t\in[0,1]}$, which may be continuous or discrete. Sampling is factorized across modalities, but training occurs in a single model with multiple heads, minimizing a weighted sum of modality-specific losses (A4).

Tasks. A *task* τ is defined by a partition of modalities,

$$\mathcal{M} = \mathcal{M}^\tau_{\text{gen}} \cup \mathcal{M}^\tau_{\text{cond}} \cup \mathcal{M}^\tau_{\text{abs}},$$

where generated modalities $\mathcal{M}^\tau_{\text{gen}}$ are modeled via flow matching, conditioned modalities $\mathcal{M}^\tau_{\text{cond}}$ are fixed conditioning information for the task, and absent modalities $\mathcal{M}^\tau_{\text{abs}}$ are not modeled in the system. For each $m \in \mathcal{M}^\tau_{\text{gen}}$, the task specifies:

- a prior distribution p_0^m (source distribution),
- a coupling π^m with the data distribution,
- a conditional probability path $p_t^m(x|x_0, x_1)$ defining the flow.

This ties the task formalization directly to multi-modal flow matching: generated modalities evolve under their own flows, while conditioning ensures compatibility across modalities.

OMTRA Instantiation. An instantiation of OMTRA is parameterized by a set of tasks

$$\mathcal{T} = \{\tau_1, \ldots, \tau_K\}.$$

If $|\mathcal{T}| > 1$, parameters are shared across tasks, yielding a multi-task model. Each $\tau \in \mathcal{T}$ corresponds to a distinct generative problem—e.g., unconditional *de novo* ligand design, pocket-conditioned design, or docking—yet all are unified by the same multi-modal flow matching framework. A comprehensive list of tasks and modalities currently supported by OMTRA are provided in Appendix A8.

A5 Data Processing and Representation

We use multiple datasets but apply consistent data processing and define a consistent molecular representation, so that OMTRA can be trained across all of them. Here we describe our representation of biomolecular systems as seen by the OMTRA model.

A5.1 Ligands

Molecules were filtered if they contained elements outside of a pre-determined supported set (C, H, N, O, F, P, S, Cl, Br, I, and B). Molecules are sanitized by rdkit. Following prior work [12, 27], all molecules are kekulized. Molecules that fail to be sanitized and kekulilzed are excluded from the dataset. Atoms were stripped of hydrogens via rdkit; notably this function will preserve some hydrogens in cases that they are essential for preserving molecule identity. We store bond orders between all pairs of ligand atoms as discrete edge features. The additional essential features to encode ligand identity are the element of every atom and its formal charge. Le et al. [50] demonstrated that incorporating other atom-level descriptions such as hybridization as additional modalities in the generative process can improve performance. Towards this end, for every ligand atom we use rdkit to compute the number of implicit hydrogens, aromaticity, hybridization state, ring membership, and chirality for each atom. However, rather than defining seven discrete modalities on ligand atoms, we condense them all down to one discrete modality for atom type; we refer to this technique as "condensed atom typing". Specifically, we search our datasets for all unique observed tuples of (atom element, formal charge, number of implicit hydrogens, aromaticity, hybridization state, ring membership, chirality), and use this set of observed values to define our "vocabulary" for assigning ligand atoms to discrete types. Thus we only model one discrete modality on every ligand atom. The advantage of condensed atom typing is that (i) atom types are more informative and correlated with 3D geometry and local topological structure than if we used the strictly necessary modalities of atom element + formal charge and (ii) there are fewer modalities to jointly sample, leaving less room for error. We find that training models for conformer generation and de novo design with condensed atom typing gives better performance than either 1. defining all the discrete ligand atom features as separate modalities or 2. only using atom type and formal charge as the ligand atom typing modalities.

A5.2 Pharmacophores

Pharmacophores are abstract representations of the molecular features responsible for key interactions with biological targets. In our framework, pharmacophore features are identified using SMARTS-based pattern matching via RDKit, with a predefined dictionary of SMARTS patterns corresponding to commonly recognized feature types.

The pharmacophore features extracted include:

- **Aromatic rings:** five- and six-membered aromatic ring systems.
- **Hydrogen bond donors:** nitrogen, oxygen, and sulfur atoms bonded to one or more hydrogen atoms.
- **Hydrogen bond acceptors:** nitrogen and oxygen atoms capable of accepting hydrogen bonds, excluding those involved in conjugation or resonance.
- **Hydrophobic groups:** nonpolar regions such as alkyl chains and hydrophobic ring systems, including certain sulfur-containing groups.
- **Halogens:** fluorine, chlorine, bromine, and iodine atoms bonded to carbon atoms.
- **Positively charged groups:** atoms or functional groups carrying a formal positive charge.
- **Negatively charged groups:** atoms or functional groups carrying a formal negative charge.

For each matched SMARTS pattern, the 3D coordinates of the corresponding atoms are obtained from the ligand's conformer, and the centroid of these atoms defines the pharmacophore feature's spatial location. From a modeling perspective, each pharmacophore feature is a point in Cartesian space with an associated discrete type. Thus, pharmacophore features are represented as nodes in graphs, and two modalities (position, type) are associated with pharmacophore nodes.

In the context of a protein binding pocket with a reference ligand, only pharmacophore features of the reference ligand within a cutoff distance of compatible pocket atoms (i.e. those likely to interact) are retained, and a random subset of 1–8 features is selected to be retained as conditioning information. For tasks without a protein pocket, sub-sampling is performed from the full set of available pharmacophore features. This enables flexible pharmacophore-conditioned generation in both pocket-aware and pocket-free settings, where the model is guided either by potential protein–ligand interactions or by the spatial distribution of ligand functional groups.

A5.3 Protein Pockets

Water, hydrogens, and deuterium atoms were stripped from protein structures. Binding pockets were extracted by selecting residues with at least one atom within 8Å of a ligand atom.

Since our graph neural network operates on all-atom representations of molecular structure, we employ a few techniques to enable OMTRA to better reason about the molecular structure beyond just a ball of atoms. On every protein atom node there is an associated discrete modality for the chemical element. There is also a discrete modality for the atom name, which uniquely identifies the atom of each canonical amino acid. This enhances the model's ability to distinguish backbone carbons from sidechain carbons, for example.

We also include a discrete modality on protein atom nodes indicating the type of residue that the atom belongs to. We support non-canonical amino acid types while still keeping the discrete vocabulary small by mapping non-canonical amino acids to their closest canonical amino acid. We use a pre-defined substitution map obtained from pdbfixer [51].

Finally, we include a residue positional encoding that is the sinusoidal positional encoding from Vaswani et al. [52] applied to the residue index, broadcasted out to each atom belonging to that residue. This feature was critical for enabling flexible-protein tasks but not for the tasks presented / evaluated in this paper.

A5.4 NPNDEs

Protein complexes in the PLINDER and Crossdocked dataset may contain multiple small, non-protein molecules. We distinguish between ligands (designable elements of interest) and non-designable, non-protein entities (NPNDE). All small molecules were retained in context, but models were not trained to generate non-designable elements. These were defined as: experimental artifacts, cofactors, ions, covalently bonded saccharides, molecules with > 120 heavy atoms or unresolved atoms, molecules forming more than one covalent bond to the protein, or molecules interacting with fewer than one protein residue.

NPNDE representation is same as ligands but the NPNDE graph is not fully connected. We also do not perform condensed atom typing, nor do we store extra features required for condensed atom typing.

A5.5 Linked Apo Structures

For PLINDER entries linked to unbound or AlphaFold-predicted structures, we first superposed the linked structure onto the bound complex prior to pocket extraction. We store 3 copies of the Plinder dataset; all plinder systems, plinder systems with exerimental apo linked structure, and plinder systems with AlphaFold-predicted linked structures.

A6 Datasets

A6.1 Pharmit

Pharmit [53] is an open-source web application for rapid screening of pharmacophores against databases of conformers using the pharmer algorithm [54]. As such, since its implementation, its creator has been downloading smiles strings of publicly available molecule databases and using rdkit ETKDG[55] + UFF minimization [56] to generate 3D conformers [57], so as to enable pharmacophore screening against these databases.

We converted every molecule on the Pharmit production server (that passed filtering criteria) into regular numpy arrays stored in Zarr. We storm the atom type, formal charge, chirality, hybridization, aromaticity, and the number of implicit hydrogens on every atom, as well as the bonding topology.

The Pharmit Dataset is comprised of following major public and commercial libraries: ChEMBL34, ChemDiv, ChemSpace, Enamine, MCULE, MCULE-ULTIMATE, MolPort, NCI Open Chemical Repository, PubChem, LabNetwork, and ZINC. This aggregation provides broad coverage of drug-like chemical space across academic, government, and vendor-curated collections. Molecules were filtered if they contained elements outside of a pre-determined supported set and if they could not be sanitized and kekulized by rdkit, resulting in a dataset of more than 500 million unique molecules. We retain only 1 conformer per molecule, the lowest energy conformer obtained by our conformer generation method.

A6.2 Plinder

The PLINDER dataset comprises over 400,000 annotated protein-ligand interaction systems curated from the Protein Data Bank [34]. A subset of systems also contain paired unbound and/or predicted structures. PLINDER additionally provides training/evaluation splits based on their novel approach to minimize leakage by leveraging similarity metrics at the level of the protein, pocket, interaction, and ligand.

To ensure high structural quality, we applied multiple filters to the PLINDER dataset to ensure OMTRA was only trained on useful examples. Systems were excluded if they had crystallographic resolution worse than 3.5 Å, $R > 0.40$, $R_{\text{free}} > 0.45$, or $|R - R_{\text{free}}| > 0.075$. Additionally, we leveraged the PoseBusters metadata to remove systems in which ligands exhibited a volume overlap greater than 0.075 with the protein or cofactors.

We computed the fraction of ligand atoms coming within 4Å of a protein pocket atom for each system; systems having fewer than 35% of ligand atoms in contact with protein atoms were also filtered from the final dataset. Systems may fail to meet this threshold due to interactions with membranes or symmetry mates, which are not modeled by OMTRA.

A6.3 Crossdocked

The Crossdocked dataset [35], derived from experimentally determined protein-ligand complexes in PDBbind [58], comprises of over 22.5 million protein-ligand poses generated by systematically docking ligands into both cognate and noncognate receptors, grouped by binding site similarity [35]. Through docking 13,780 unique ligands to multiple similar binding pockets, the dataset captures realistic variability in protein conformation and ligand orientation, enabling the training and evaluation of models across multiple structure-based tasks.

To reduce overlap between similar protein structures and chemotypes in the training and test sets, the original Crossdocked2020 paper adopted clustered cross-validation splits for both PDBBind and Crossdocked data. For the PDBBind datasets, clusters were defined by grouping receptors with >50% sequence identity, or with >40% sequence identity and >90% ligand similarity, as computed using RDKit fingerprints. Under this scheme, highly similar ligands are only assigned to different clusters if the corresponding receptors share less than 40% sequence identity, reducing information leakage across splits [35]. For the CrossDocked2020 dataset, clustering was based on Pocketome-defined pockets, further grouped using the ProBiS structural alignment algorithm with a z-score threshold of 3.5, enabling cross-validation across structurally similar binding sites [35].

However, in this work we adopt a different split scheme, introduced by Luo et al. (2022). These splits were generated by clustering data at 30% sequence identity, and randomly drawing 100,000 protein-ligand pairs for training, and 100 proteins from remaining clusters for testing [36, 41].

Although these splits are less rigorous than those of the original Crossdocked authors and likely contain information leakage, they are widely reused. As such, we use these splits to enable standardized evaluation across models.

A6.4 Posebusters Benchmark Set

The PoseBusters Benchmark set is a dataset of 428 diverse protein-ligand complexes originally curated by Buttenschoen et al. [37]. This is a moderately difficult docking benchmark comprising 428 diverse protein-ligand complexes. We use this dataset to compare OMTRA's re-docking performance to existing models. We sample 40 docked poses for each system and rank OMTRA-docked poses using the Autodock Vina [42] scoring function. Notably in our evaluations of the PoseBusters benchmark set, we did not take care to remove overlapping examples from the Plinder training set, and as such our estimates of docking performance may be optimistic.

A7 Flow Matching Formulation

Our flow matching formulation closely follows that of Dunn and Koes [12]. Continuous and discrete modalities, and the multi-modal flow matching formulations are nearly identical; there are just more modalities supported under OMTRA.

A7.1 Introduction

Flow matching [23, 25, 22, 24] prescribes a method to interpolate between two distributions q_{source} and q_{target} by modeling a family of intermediate distributions $\{p_t\}_{t\in[0,1]}$ with $p_0 = q_{\text{source}}$ and $p_1 = q_{\text{target}}$. Samples evolve along conditional probability paths defined by a coupling $p(x_0, x_1)$ between initial and final states. Given a conditioning variable $z = (x_0, x_1)$, the marginal path satisfies

$$p_t(x) = \mathbb{E}_{p(z)}[p_t(x|z)] \tag{A1}$$

with the property that $x_t \sim p_t(x|x_0, x_1)$ can be simulated without iterative sampling.

Continuous Flow Matching For continuous variables $x \in \mathbb{R}^d$, the marginal process is sampled by a vector field $u_t(x) = \frac{dx}{dt}$. The vector field may be learned directly, or indirectly by predicting the conditional expectation of the final state $\mathbb{E}[x_1|x_t]$. Training then reduces to denoising: sampling $x_t \sim p_t(x|x_0, x_1)$ and minimizing error between the network prediction $\hat{x}_1(x_t)$ and the true x_1:

$$\mathcal{L}_{\text{EFM}} = \mathbb{E}_{t,p(x_0,x_1),p_t(x_t|x_0,x_1)}\left[(1-t)^{-2}\left\|\hat{x}_1(x_t) - x_1\right\|^2\right] \tag{A2}$$

Discrete Flow Matching For discrete data $x \in \mathcal{V}^N$ (sequences of tokens from a vocabulary $\mathcal{V}$), trajectories are governed by continuous-time Markov chains (CTMCs). Each token remains in its current state until it jumps to a new one. In masked discrete flow matching [13], the prior is a sequence of mask tokens, and the marginal CTMC is parameterized by a denoiser that estimates unmasked states in partially-masked sequences. Training uses a cross-entropy objective:

$$\mathcal{L}_{\text{CE}} = \mathbb{E}_{t,p(x_0,x_1),p_t(x_t|x_0,x_1)}\left[-\sum_{i=1}^{N}\log p^{\theta}_{1|t}(x_1^i \mid x_t)\,\delta_M(x_t^i)\right] \tag{A3}$$

Multi-Modal Flow Matching Data structures often involve multiple modalities $\mathcal{M}$, e.g., continuous positions and discrete types in molecular systems. Multi-modal flow matching [12, 13] extends the framework by factorizing conditional paths across modalities. A single neural network with multiple heads predicts outputs for each modality, with the overall loss being a weighted sum over modalities:

$$\mathcal{L} = \sum_{m\in\mathcal{M}} \lambda_m \mathcal{L}_m \tag{A4}$$

For OMTRA, we use the denoising loss (A2) and cross-entropy loss (A3) for continuous and discrete modalities, respectively.

A7.2 Design Choices for Continuous Flow Matching

Prior and Coupling For continuous data $x \in \mathbb{R}^d$ (atom positions), our prior is a Gaussian distribution with mean equal to the center-of-mass of the ground-truth ligand and variance of 2.573Å which is the mean variance of ligand atomic coordinates in the data distribution. For conformer generation or docking, where the ligand identity is conditioning information, the coupling distribution for atomic coordinates is the independent coupling. For *de novo* ligand generation, the coupling distribution involves a distance-minimizing permutation of (initial,final) atom pair assignments as proposed in Dunn and Koes [12].

Conditional Probability Path The conditional probability path is generated implicitly by the geometry distortion interpolant proposed by Dunn and Koes [12]:

$$X_t = (1-t)\,X_0 + t\,X_1 + \mathbb{I}[t \geq t_{\text{distort}}]\,\big(M \odot \varepsilon\big), \tag{A5}$$

where $\mathbb{I}$ is the indicator function, $\odot$ is the Hadamard product, $M \in [0,1]^N$ is a binary mask over atoms having the property $M_i \sim \text{Bernoulli}(p_{\text{distort}})$, and $\epsilon \in \mathbb{R}^{N\times 3}$ is a per-atom displacement having the property $\epsilon_i \sim \mathcal{N}(0, \sigma_{\text{distort}} I_3)$. Geometry distortion is controlled by three hyperparameters that are set to $p_{\text{distort}} = 0.2$, $t_{\text{distort}} = 0.5$, and $\sigma_{\text{distort}} = 0.5$.

A7.3 Design Choices for Discrete Flow Matching

For all discrete modalities generated by OMTRA, we employ discrete flow matching (DFM) as developed by Campbell et al. [13] and Gat et al. [32]. Specifically this is masked discrete flow matching where the prior is a sequence of a special mask state. At inference time, individual tokens can jump between mask tokens, unmasked tokens, and back to masked tokens. The rate at which remasking happens can be controlled by an inference-time hyperparameter η. The training objective is as described by (A3); our model produces, for each node/edge, logits over possible discrete states at $t = 1$ given the current state at time t. For details on the conditional probability paths and sampling techniques, we refer the reader to Dunn and Koes [12].

A8 Supported Tasks and Modalities

OMTRA supports a variety of modalities, each representing a distinct type of input features for a particular task. Modality examples include ligand and protein structure, identity, pharmacophore types, and so forth. A complete list of modalities used in OMTRA is outlined in Table A2. These modalities are grouped by conceptual function and are used as building blocks across multiple generative tasks.

OMTRA also supports a variety of tasks, each representing a different generative or predictive objective. Tasks specify which entities are present and determine how modalities are used, either as constraints or generated outputs. Task examples include de novo ligand design, pocket and pharmacophore conditioned de novo design, and rigid docking. A complete list of currently supported and upcoming tasks is outlined in Table A2 and Table A4 respectively.

A8.1 Supported Modalities

Table A2: OMTRA Supported Modalities. The modality `Ligand Condensed Features` includes implicit hydrogens, aromaticity, hybridization, in ring, and chirality, all represented as categorical variables. Condensed atom typing is described in detail in Section A5.1.

Modality	Group	Data Type	Graph Entity
Ligand Atom Positions	Ligand Structure	Continuous	Node
Ligand Condensed Atom Features	Ligand Identity	Discrete	Node
Ligand Bond Orders	Ligand Identity	Discrete	Edge
Pharmacophore Positions	Pharmacophore	Continuous	Node
Pharmacophore Types	Pharmacophore	Discrete	Node
Protein Atom Positions	Protein Structure	Continuous	Node
NPNDE Positions	Protein Structure	Continuous	Node
Protein Residue Names	Protein Identity	Discrete	Node
Protein Atom Elements	Protein Identity	Discrete	Node
Protein Atom Names	Protein Identity	Discrete	Node
NPNDE Atom Elements	Protein Identity	Discrete	Node
NPNDE Charges	Protein Identity	Discrete	Node
NPNDE Bond Order	Protein Identity	Discrete	Edge

A8.2 Supported Tasks

Table A3: OMTRA Supported Tasks

Task	Entities Present	Modality Groups Fixed	Modality Groups Generated
Unconditional De Novo Ligand Design	Ligand	None	Ligand Identity, Ligand Structure
Ligand Conformer Design	Ligand	Ligand Identity	Ligand Structure
Rigid Docking	Ligand & Protein	Ligand Identity, Protein Identity, Protein Structure	Ligand Structure
Pocket-Conditioned De Novo Ligand Design	Ligand & Protein	Protein Identity, Protein Structure	Ligand Identity, Ligand Structure
Pharmacophore-Conditioned De Novo Ligand Design	Ligand & Pharmacophore	Pharmacophore	Ligand Identity, Ligand Structure
Pharmacophore-Conditioned Ligand Conformer Design	Ligand & Pharmacophore	Ligand Identity, Pharmacophore	Ligand Structure
Pharmacophore-Conditioned Rigid Docking	Ligand, Protein, & Pharmacophore	Protein Identity, Protein Structure, Ligand Identity, Pharmacophore	Ligand Structure
Pocket and Pharmacophore-Conditioned De Novo Ligand Design	Ligand, Protein, & Pharmacophore	Protein Identity, Protein Structure, Pharmacophore	Ligand Identity, Ligand Structure

A8.3 Upcoming Tasks

Table A4: OMTRA Upcoming Tasks

Task	Entities Present	Modality Groups Fixed	Modality Groups Generated
Predicted Apo, Flexible Protein De Novo Ligand Design	Ligand & Protein	Protein Identity	Ligand Identity, Ligand Structure, Protein Structure
Flexible Protein Docking	Ligand & Protein	Ligand Identity, Protein Identity	Ligand Structure, Protein Structure
Flexible Protein and De Novo Ligand Design	Ligand & Protein	Protein Identity	Protein Structure, Ligand Identity, Ligand Structure
Experimental Apo-Conditioned, Flexible Protein Docking	Ligand & Protein	Ligand Identity, Protein Identity	Ligand Structure, Protein Structure
Predicted Apo-Conditioned, Flexible Protein Docking	Ligand & Protein	Ligand Identity, Protein Identity	Ligand Structure, Protein Structure
Unconditional De Novo Ligand and Pharmacophore Design	Ligand & Pharmacophore	None	Ligand Identity, Ligand Structure, Pharmacophore
Flexible Protein and De Novo Pharmacophore Design	Protein & Pharmacophore	Protein Identity	Protein Structure, Pharmacophore
Experimental Apo-Conditioned, Flexible Protein and De Novo Pharmacophore Design	Protein & Pharmacophore	Protein Identity	Protein Structure, Pharmacophore
De Novo Protein, Ligand, and Pharmacophore Design	Ligand, Protein, & Pharmacophore	Protein Identity	Protein Structure, Ligand Identity, Ligand Structure, Pharmacophore

A9 Architecture

Our architecture is essentialy the simplest possible extension of FlowMol3 [12] to heterogeneous graphs with type-specific message passing and feature updates. Nodes $i \in V$ carry Cartesian positions $x_i \in \mathbb{R}^3$, scalar features $s_i \in \mathbb{R}^{d_s}$, and vector features $v_i \in \mathbb{R}^{d_v \times 3}$. Edges $(i, j) \in E$ are typed; ligand–ligand and other–other edges additionally carry scalar features $e_{ij} \in \mathbb{R}^{d_e}$. Operations on (x_i, v_i) are SE(3)-equivariant; operations on (s_i, e_{ij}) are SE(3)-invariant. Node vectors v_i are first-order geometric vectors relative to x_i. SE(3)-equivariant operations are enabled by a custom variant of GVPs [33] introduced in Dunn and Koes [12].

A9.1 Initial Feature Embeddings

All nodes in the system are initialized with scalar embeddings of a fixed latent size (256). Additionally, ligand-ligand and NPNDE-NPNDE edges also have initial scalar edge embeddings.

Scalar node embeddings are initialized by the node modalities defined on that node type that are not positions. For ligand atoms, these are the condensed atom types. We use a standard embedding lookup table (pytorch's nn.Embedding module) to map each discrete type to a learnable fixed-length vector. The only continuous modality on node features that is not a position is the residue positional embedding described in Section A5.3. The modality embeddings are concatenated together along with a sinusoidal time embedding and a learnable task embedding. These features are passed through a scalar embedding MLP that is unique to each node type to generate intital scalar node embeddings.

Ligand-ligand and NPNDE-NPNDE edges are simply embedded with an nn.Embedding lookup of the bond order on that edge.

A9.2 Graph Convolutions

Message Passing. Messages are generated per edge type $r \in \mathcal{T}_E$:

$$m^r_{i \leftarrow j} = \phi_r(s_i, s_j, v_i, v_j, e_{ij}, x_i - x_j), \tag{A6}$$

where ϕ_r is an edge-type–specific function. Each node aggregates all incoming messages from all edge types:

$$M_i = \sum_{r \in \mathcal{T}_E} \sum_{j \in \mathcal{N}_r(i)} m^r_{i \leftarrow j}. \tag{A7}$$

A9.3 Node and Edge Feature Updates

Each node type $\alpha \in \mathcal{T}_V$ has its own update function ψ_α:

$$(s_i, v_i) \quad \mapsto \quad \psi_{\alpha(i)}(s_i, v_i, M_i), \tag{A8}$$

which updates only scalar and vector features. Node positions x_i are not inputs to ψ_α and are updated separately. We defer the reader to Dunn and Koes [12] for the precise node-update step equation.

Graph convolutions (message passing + node feature updates) are interleaved with:

- **Node position updates:** node-wise GVPs operating on (s_i, v_i, x_i) to produce new x_i.
- **Edge feature updates:** edge-wise MLPs that take $(e_{ij}, s_i, s_j, x_i, x_j)$ to produce new e_{ij}.

A9.4 Block Structure

Two graph convolutions followed by position and edge updates form one *ConvolutionBlock*; OMTRA stacks 4 such blocks.

A9.5 Model Outputs and Multi-Tasking

For node positions being generated, final x_i are taken directly from the last *ConvolutionBlock* output as these blocks update node positions internally. The final latent node/edge features are mapped to categorical logits by shallow MLP heads. If a graph lacks nodes or edges of a given type, the corresponding ϕ_r or ψ_α is simply skipped. Node-position updates and categorical logit heads are applied only when the relevant modality is being generated under the current task.

A9.6 Message Generation

Here we describe the specific form our message-generating functions (A6). The message generating functions are chains of GVPs [16]. GVPs accept and return a tuple of scalar and vector features. Therefore, scalar and vector messages $m^{(s)}_{i \to j}$ and $m^{(v)}_{i \to j}$ are generated by the message-generating function ϕ_M, which is 3 GVPs chained together.

$$m^{(s)}_{i \to j}, m^{(v)}_{i \to j} = \phi_M \left(\left[s^{(l)}_i : e^{(l)}_{ij} : d^{(l)}_{ij} \right], \left[v_i : \frac{x^{(l)}_i - x^{(l)}_j}{d^{(l)}_{ij}} \right] \right) \tag{A9}$$

where : denotes concatenation and d_{ij} is the distance between nodes i and j at molecule update block l. In practice, we replace all instances of d_{ij} with a radial basis embedding of that distance before passing through GVPs or MLPs.

A10 Protein-Task Evaluation Pipeline

A10.1 De Novo Evaluation Methods

We benchmark OMTRA against four models for pocket-conditioned, de novo ligand design: DrugFlow, a flow matching model; DiffSBDD and TargetDiff, two diffusion models; and Pocket2Mol, an autoregressive model [41, 28, 40, 39]. We evaluated the models on the Crossdocked test split published by Luo et al. [59]. For each protein pocket, we sample 100 ligands.

The PoseBusters suite is widely used to evaluate the chemical and geometric plausibility of small-molecule ligands and their interactions with protein binding pockets [37]. In the context of pocket-conditioned *de novo* design, PoseBusters checks whether the ligand can be sanitized by RDKit, has reasonable geometry, is not fragmented, has reasonable internal ligand energy, has minimal clashes, and has reasonable proximity to other molecules in the system such as the protein and co-factors. We define the metric %PB-Valid as the fraction of ligands that pass all PoseBusters checks. Ligands that cannot be sanitized are automatically marked as not PB-Valid, and other individual tests implemented by posebusters are not performed.

We use Gnina [44] to obtain scores from Autodock Vina [42] (this amounts to running Gnina with the –score-only option).

The strain energy of the generated ligand measures how much internal energy is stored in the ligand, with a lower strain generally corresponding to more favorable binding interactions with the pocket. It is computed by comparing the difference in energy before and after energy minimization. Relaxation and energy computation were performed by PoseCheck using the Universal Force Field (UFF) [56].

Interaction fingerprints were extracted using PoseCheck [38] which uses ProLIF [60] to identify different protein-ligand interactions based on smarts patterns and geometries [38, 60]. PoseCheck reports, for each interaction type, the number of interactions occurring in a protein-ligand complex divided by the number of ligand atoms. We use these normalized interaction counts to compute the metrics "interaction parity" and "interaction recovery". The interactions types we use for these analyses are hydrogen bond donors, hydrogen bond acceptors, and hydrophobic interactions.

Interaction parity measures whether the number of interactions occurring between a generated ligand and a target protein are greater than or equal to that of the ground-truth ligand. Interaction parity is a binary value, a pass/fail test, performed on each generated ligand. For each interaction type, we check if the normalized interaction count is greater than or equal to the normalized interaction count for the ground-truth ligand. Interaction parity is "true" for a single ligand if the normalized interaction counts are greater than that of the ground-truth ligand for all interaction types. The "% Interaction Parity" reported in the main body are the fraction of sampled ligands that have interaction parity.

"Interaction recovery" is defined as when when the generated ligand reproduces the same types of interactions with the same residue as the ground truth ligand. The interaction recovery rate is the fraction of interactions made by the ground truth ligand that are recovered by the *de novo* generated ligand. These interaction-based metrics give insight into whether the models reproduce the specific interactions observed in known binding ligands.

For tasks that include pharmacophores, we report the additional metric "% Pharm Matches", which is the fraction of pharmacophores in the ground truth ligand that are matched by a pharmacophore in the generated ligand. A match requires both the correct pharmacophore identity as well as the right location, defined as being within 1Å of the ground truth pharmacophore.

A10.2 Docking Evaluation Methods

PoseBusters was also used to evaluate the docked poses. In its docking evaluation mode, PoseBusters includes the checks described for *de novo* design evaluation together with additional analyses that use the ground truth ligand conformation. These assess whether the molecular formula, bonds, tetrahedral chirality, and double bond stereochemistry of the generated ligand match the ground truth, as well as whether the RMSD between the generated and the ground truth conformation is <2Å . We report how frequently docked poses are <2Å from the ground truth. We also report %PB-Valid which is the fraction of docked poses that are within <2Å RMSD from the ground-truth and pass all of the PoseBusters physical plausibility checks.

To be consistent with prior works [37, 7, 61, 43], we report docking metrics on the "Top N" sampled poses. This requires sampling multiple docked poses, and then ranking them by some metric. After ranking poses amd selecting the top-N, we compute: (i) the fraction of systems with at least

one pose in the top N whose RMSD is <2Å and (ii) the fraction of systems with at least one pose among the top-N whose RMSD is <2Å and is PB-Valid. We report these metrics for N=1 and N=5.

Each of the baseline docking methods evaluated has its own ranking method; the deep-learning based docking methods have their own confidence modules for ranking, and conventional docking methods use the same scoring function to rank poses as was used to guide MCMC sampling of poses. OMTRA poses are ranked by a scoring function that is the Autodock Vina score augmented with an additional penalty for violation of chirality. This practice is consistent with AlphaFold3 [7] which ranks poses by a composite score including the confidence and penalties for clashes and chirality violations.

For tasks with pharmacophore conditioning, we again report the metric "% Pharm Matches". Here, since ligand identity is fixed, the metric simply measures the degree to which the pharmacophores of the generated ligand conformation are aligned with those of the ground truth ligand. We also report interaction recovery as described in the preceding section.