FlexiDock: Compositional diffusion models for flexible molecular docking

Zichen Wang AWS AI Research & Education zichewan@amazon.com

Zhengyuan Shen AWS AI Research & Education donshen@amazon.com Balasubramaniam Srinivasan AWS AI Research & Education srbalasu@amazon.com

George Karypis AWS AI Research & Education gkarypis@amazon.com

Huzefa Rangwala AWS AI Research & Education rhuzefa@amazon.com

Abstract

Molecular docking is a critical process in structure-based drug discovery to predict the binding conformations between a protein and a small molecule ligand. Recently, deep learning-based methods have achieved promising performance over traditional physics-based search-and-score methods. Despite their success on accurately predicting the binding poses of the small molecule ligands, modeling of protein flexibility and dynamics still remains largely unexplored for docking. We observe that models that do not account for the protein flexibility suffer a large performance drop in cases where proteins undergo large conformational changes upon ligand binding. To address this gap, we developed FlexiDock, a compositional alternating neural diffusion process, which include two diffusion models to explicitly model the conformational flexibilities of proteins and ligands, respectively. The compositional diffusion process is inspired by the induced-fit model in flexible docking. We found the compositional diffusion is able to improve the structural prediction of the proteins upon ligand binding. Our method also offers promising insights into modeling proteins' conformational switches.

1 Introduction

The physical interactions between proteins (receptor) and their binding partners (ligands) can be modulated by drugs in the forms of small molecule ligand, peptides and antibodies. Molecular docking computationally predicts the preferred position, orientation, and conformation of one molecule to a second when a ligand and a target are bound to each other to form a stable complex under ambient conditions. Therefore, molecular docking is critical for drug discovery: an efficient and accurate docking algorithm can speed up the development of a potential drug candidate for a biological target.

Recently, many deep learning based methods such as EquiBind [27], DiffDock [5] have been developed for blind docking, where the models do not assume any prior knowledge about the binding pocket. Although DiffDock achieved SoTA performance on docking small molecule ligands to the bound (*holo*) structure of proteins, it still performs relatively poorly when the input protein conformation is the unbound (*apo*) structures. Since holo structures are unavailable, drug discovery scientists usually have the need to perform blind docking for newly designed ligands with *apo*

Machine Learning for Structural Biology Workshop, NeurIPS 2023.

structures. The drastic performance drop in the apo-setting is due to absence in explicitly modeling the proteins' conformational flexibilities, and is the key limitation of rigid docking compared to flexible docking [2]. Unfortunately, DiffDock [5] cannot be extended trivially to jointly model the receptor and ligand flexibilities as the decomposition over the product spaces used in DiffDock is no longer applicable.

To alleviate this, we developed FlexiDock, a compositional diffusion process containing two diffusion models to explicitly model the conformations of the small molecule ligand and protein receptor, respectively. The reverse diffusion process of our models can be used in flexible docking setting between an apo-structure of a protein and a small molecule ligand. Our model draws inspiration from the classic induced-fit theory [12], which states the binding of ligand can induce the protein conformation changes so that the overall energy of the binding complex is minimized.

The main contributions of this work are:

- 1. formulated composition of two independently trained diffusion models
- 2. developed a ligand-conditioned receptor (LCR) score module to model the conformation changes of proteins using conditioned diffusion model
- 3. developed a truncated reverse diffusion process for the LCR score model at inference phase to make fine-grained adjustment to protein conformation
- 4. achieved improved performance in blind docking in apo-setting on the PDBBind dataset and showed promises on modeling proteins' apo-holo conformational switches

2 Related work

Rigid-body blind docking. In this setting, protein receptor is assumed to be rigid without conformational flexibility. The ML model is tasked to identify the binding pocket on the receptor and predict a pose for a given small molecule ligand. Models such as EquiBind [27] and TANKBind [17] are regression-based methods leveraging SE(3)-equivariant neural networks to predict the key points or interatomic distance matrix of the ligand and the receptor. DiffDock [5] formulates this task as a diffusion generative modeling method: the diffusion process modify the ligand's relative position to the receptor by roto-translation, and conformation by rotating the torsional bond angles. As they disregard the flexibilities of the protein receptors, the holo conformation of the receptor is typically used in rigid-body setting. It has been shown that the docking performances of decrease significantly with apo-structures as inputs [30, 5].

Molecular docking via co-folding. With the seminal success of protein folding algorithms like AlphaFold2 [10], these methods tackle the molecular docking challenge using a receptor-ligand co-folding approach. For instance, DPL [19] and NeuralPLexer [22] solve the structure of the receptor-ligand complex from a protein sequence and a ligand's structure. These models modify the neural architectures used for protein folding to incorporate the receptor-ligand binding geometry, such as contact prediction, to achieve the accurate prediction of receptor-ligand complex structures. NeuralPLexer can also be applied in both holo and apo blind docking settings and has been shown to outperform DiffDock in the holo-setting [22]. However, both settings fail to model the receptor flexibilities from the initial conformation.

Diffusion models for protein structures Diffusion generative models [24, 26, 6] has shown great promise in modeling proteins' 3D structures, with applications ranging from full suites of protein design problems [7, 29], to specific problems such as side-chain packing [33]. Some diffusion models for protein formulate their diffusion process in the Cartesian space using the backbone atoms [28] or amino acid residue frames [15, 32]. FoldingDiff's [31] diffusion process works exclusively in modeling the backbone angles, which can be used to reconstruct the 3D backbone structures. [1] uses a combination of Cartesian coordinates and internal angels. These protein structural diffusion models make gradual changes to the protein conformations and have shown promise in modeling multiple conformations for a given protein [9]. In particular, SBalign [25] explicitly predicts the conformational changes between apo and holo states of proteins using diffusion Schrödinger Bridges, a diffusion process designed to interpolate between two modes of data. However, they have not been applied in the flexible docking setting.



Figure 1: Overview of FlexiDock. *Left*: in the training phase, we train two independent diffusion models responsible for modeling the conformational flexibilities of the receptor and ligand, respectively. The two forward diffusion processes add noise to the angles of the receptor backbone and translates, rotates, and updating the torsional angles of the ligand. *Right*: in inference phase, we alternate the reverse diffusion processes to recover the position and conformation of the ligands, then to optimize the conformation of the receptor.

3 FlexiDock

Notations: We use Ω_R and Ω_L to denote the continuous sample spaces associated with receptors and ligands, respectively. Let R, L denote random variables associated with receptor and ligand, respectively, and let $P(R|L = l \in \Omega_L)$ and $P(L|R = r \in \Omega_R)$ denote the respective conditional probability distributions. The diffusion models in the next sections independently model the conditional probability distributions given above via the Continuous State Markov Chains $MC_{R|L}$ and $MC_{L|R}$ (for state transitions) respectively, and we use $s_{\phi_1}(r; l, t)$ and $s_{\phi_2}(l; r, t), r \in \Omega_R, l \in \Omega_L, t \in [0, 1]$, to denote the corresponding score predictor neural networks given the ligand and receptor, respectively.

From the perspective of ligand and receptor interactions, the conditional distributions $P(R|L = l \in \Omega_L)$ and $P(L|R = r \in \Omega_R)$ can be seen to model conformational selection (Figure 2b bottom left) and ligand fitting (Figure 2b top right), whereas the joint distribution P(L, R) models the mutual fit of ligand and receptor (Figure 2b bottom right).

Overview: FlexiDock is composed of two independently trained score-based diffusion generative models [26], each responsible for modeling the conformational flexibilities of the receptor r and ligand l, respectively (Fig. 1). Notably, the two diffusion processes only add noises to the molecule of interest while keeping its binding partner's relative position and conformation fixed.

We adopt the score model and diffusion process from DiffDock [5] to model the translation, rotation, and torsion angles of the small molecule ligands conditioned on the protein backbones. The forward diffusion process applies random roto-translation transformation to the ligand and torsional updates to its rotatable bonds. The ligand score model learns to predict the three terms: $t \in \mathbb{R}^3$, $C \in \mathbb{R}^{3\times 3}$, $\theta^{(l)} \in SO(2)^m = s_{\phi_2}(l; r, t)$, corresponding to the noise updates in translation, rotation, and torsion angles at each diffusion step. Separately, we use another diffusion process to add noise to the internal angles of the receptor backbones while keeping the relative positions, orientations, and poses of the ligand fixed. Similar to the ligand score model, we train a ligand-conditioned receptor (LCR) score model (Sec. A.1) to learn the angular updates on the protein backbones: $\theta \in SO(2)^{n\times 6} = s_{\phi_1}(r; l)$.

In the inference phase, our framework is designed to work in the flexible docking setting, in contrast to rigid-body docking in holo setting (Fig. 2), we optimize the relative positions and conformations for both the ligand and receptor. To do that, we alternate the two reverse diffusion processes for ligand and receptor: the former finds the binding region on the receptor for the ligand and generates a bound pose, whereas the latter optimize the receptor to better fit for the ligand's current conformation. We term this process as alternating inference, which we proved to asymptotically converge to the the joint distribution of ligand-receptor complex comformations (Sec. A.2). We also invent a truncated reverse diffusion process to mitigate the over-correction of receptor conformations (Sec. A.3).



Figure 2: Holo vs apo settings for blinding docking. (a) the holo structure of the protein receptor is provided and remains rigid in the docking process, only the ligand's conformations and positions are optimized. (b) the conformations of both the receptor and ligands are optimized during apo flexible docking. Red arrows denote the process of optimizing ligand conformation towards a rigid receptor; blue arrows denote receptor conformational change; whereas the purpose arrow indicates an interative mutual fit process.

4 Results

4.1 Experimental setup

We conduct our experiments on the PDBBind dataset [16] and the dataset collected by Saldaño *et al.* [23]. The PDBBind dataset contains over 19,000 holo structures of receptor-ligand complexes collected from the Protein Data Bank (PDB) [3], whereas the Saldaño dataset curates 90 pairs of apo/holo PDB IDs corresponding to ligand-induced conformational change. We used the preprocessed PDBBind v2020 dataset generated in [27] for model training and evaluation. We follow the blind docking setup from [5] for both holo and apo settings. Briefly, all the docking methods receive two inputs: the ligand with predicted seed conformation from RDKit [13] and the holo or apo conformation of the protein receptor. For experiments on the PDBBind test set, the apo structures are predicted by ESMFold [15] from their primary sequences. To evaluate the generated conformations, we first align the generated receptor conformation to its holo-structure using the backbone atoms (N, $C\alpha$, and C) with the Kabsch algorithm [11], then compute the backbone RMSD for receptor and heavy-atom RMSD for ligand, respectively. Both ligand- and receptor- RMSDs use the conformations in the holo-structure of the complexes as ground truth. Additionally, we compute the TM-score [34] for the receptor conformation.

4.2 Rigid-body docking performs poorly in apo setting

To motivate flexible apo docking methods, we first examined the performance of DiffDock, the state-of-the-art rigid-body docking method, on both holo and apo settings. Consistent with [5], we also observed a drastic performance drop in terms of predicting the correct ligand conformations. The percentage of generated ligands conformations below 2Å decrease from 54.45% to 34.33% on the PDBBind dataset (Table S1). We next asked the potential causes of this drop by performing error analysis focusing on the input apo protein conformations. We found a strong correlation (Spearman's ρ =0.298) between the drop in ligand RMSD and the backbone RMSD between ESMFold predicted apo structures and holo structures (Fig. S1). It suggests that rigid-body docking methods such as DiffDock exhibit deteriorated performance when docking ligands to receptors with large conformational discrepancies between their apo and holo states. After all, rigid-body docking methods ignores the conformational flexibilities of the protein receptors. These observations indicate the need to develop a model to explicitly model the proteins' conformational flexibility to help optimize the apo structures. We hypothesize that an apo conformation optimized towards its holo counterpart will improve the docking performance for both the receptor and the ligand.



Figure 3: FlexiDock's LCR model improves apo structures to move closer to holo structures by local adjustments.

4.3 FlexiDock improve protein conformation prediction

We asked if the additional reverse diffusion process designed to optimize apo structures towards its holo counterpart can be coupled with the reverse diffusion processes for ligands. We found that FlexiDock's alternating inference is able to marginally improve the percentage of receptor conformation below the 2Å threshold (56.58% to 58.36%) (Table 1). This improvement is greater (56.58% to 60.5%) when we apply truncated inference for the receptors' reverse diffusion process. This is due to the time prediction model's ability to accurately quantify the structural differences between a pair of apo and holo conformations (Fig S2). Therefore, the time prediction model guides the reverse diffusion process to prevent over correcting the receptor conformation. Interestingly, we found FlexiDock turns to move the apo structures closer to holo structures via local adjustment, evidenced by the improvement in RMSD while keeping the TM-score almost unchanged (Fig 3).

conformations from the holo structure	es as references.
apo protein structures as input. RMS	D and TM-score are calculated using the receptor and ligand
Table 1: PDBBind flexible docking.	All methods receive a small molecule and a ESMFold-ed

K	Model	inference	Top-1 ro RMSD %<2 ↑	eceptor ba RMSD %<5↑	ackbone median RMSD↓	median TM-score↑	Top-1 li RMSD %<2↑	gand RMSD %<5↑	median RMSD↓
1	DiffDock	no alternation	56.58	80.43	1.59	0.965	32.74	52.67	4.61
	FlexiDock	alternating inference	58.36	80.78	1.72	0.952	32.74	52.67	4.61
	FlexiDock	alternating inference; w. truncation	60.50	80.78	1.47	0.965	32.74	52.67	4.61
2	DiffDock	no alternation	56.58	80.43	1.59	0.965	43.42	76.87	2.37
	FlexiDock	alternating inference	0.72	20.22	8.01	0.594	12.46	60.50	4.25
	FlexiDock	alternating inference; w. truncation	57.30	80.78	1.58	0.963	45.20	76.16	2.23

Next, we asked whether the apo structures transformed by FlexiDock's LCR model can result in better docked poses for ligands. To do that, we extend the alternating inference approach to iterate for K = 2 rounds. This procedure emulates the iterative mutual fit idea, whereby each iteration is a reverse diffusion process rather than a step in the reverse diffusion process. We found that the receptor structures optimized by LCR indeed lead to improved ligand poses by 2% on the RMSD<2% criterion (Table 1). However, the receptor conformations can hardly be improved by one more round alternating inference. We reason that one round of LCR inference already led to the convergence of the receptor conformation, any more denoising would lead to suboptimal receptor conformations. This is supported by the observation that K = 2 rounds of inferences without truncation leads to deformed receptor conformations (Table 1).

Table 2: Predicting apo-holo conformational switch on the Saldaño dataset. All methods receive a small molecule and an apo protein structures as input. RMSD and TM-score are calculated using the receptor and ligand conformations from the holo structures as references.

		Top-1 receptor backbone				Top-1 ligand		
Model	inference	RMSD	RMSD	median	median	RMSD	RMSD	median
		%<2↑	%<5↑	RMSD↓	TM-score↑	%<2↑	%<5↑	RMSD↓
DiffDock	no alternation	25.49	84.31	3.23	0.832	13.73	58.82	3.95
FlexiDock	alternating inference	0	66.67	4.53	0.667	13.73	58.82	3.95
FlexiDock	alternating inference; w. truncation	31.27	92.16	2.68	0.836	13.73	58.82	3.95

To assess how well FlexiDock can model the fold-switching between apo and holo pairs, we applied it to the Saldaño dataset [23]. Notably, the apo structures in this dataset are resolved by biochemical experiments, whereas the apo structures in the PDBBind dataset are predicted by ESMFold, many of which resemble their holo counterparts. This dataset offers the ability to assess the naturally occurring conformational switches induced by ligand binding. Consistent with the performance we observed from the PDBBind dataset, we demonstrate that FlexiDock's LCR score model with the truncated inference procedure improved both the RMSD<2% and RMSD<5% criteria by over 6% (Table 2), suggesting our method is able to simulate the ligand-induced conformational changes to some extent.

4.4 Qualitative case studies

Next, we examine a few FlexiDock-generated holo structures and contrasting them with their apo counterparts. We observed FlexiDock mostly modifies the backbone angles in the coiled regions to enable the modeling of protein domain movements (Sec. C.3).

5 Conclusion

In this work, we tackled the problem of flexible molecular docking by a compositional diffusion approach. Our approach explicitly models the conformational flexibilities of both the receptor and ligand, achieving competitive performance on recovering the complex structures. Qualitative analysis reveals that FlexiDock primarily adjust receptors' coiled regions. Our method shows promises in modeling and sampling proteins' diverse and dynamic conformations, which are crucial for many drug discovery and protein engineering applications.

References

- [1] Namrata Anand and Tudor Achim. Protein structure and sequence generation with equivariant denoising diffusion probabilistic models. *arXiv preprint arXiv:2205.15019*, 2022.
- [2] Dinler A Antunes, Didier Devaurs, and Lydia E Kavraki. Understanding the challenges of protein flexibility in drug design. *Expert opinion on drug discovery*, 10(12):1301–1313, 2015.
- [3] Helen Berman, Kim Henrick, and Haruki Nakamura. Announcing the worldwide protein data bank. *Nature structural & molecular biology*, 10(12):980–980, 2003.
- [4] Devlina Chakravarty and Lauren L Porter. Alphafold2 fails to predict protein fold switching. *Protein Science*, 31(6):e4353, 2022.
- [5] Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. *arXiv preprint arXiv:2210.01776*, 2022.
- [6] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.
- [7] John Ingraham, Max Baranov, Zak Costello, Vincent Frappier, Ahmed Ismail, Shan Tie, Wujie Wang, Vincent Xue, Fritz Obermeyer, Andrew Beam, et al. Illuminating protein space with a programmable generative model. *BioRxiv*, pages 2022–12, 2022.
- [8] Bowen Jing, Gabriele Corso, Jeffrey Chang, Regina Barzilay, and Tommi Jaakkola. Torsional diffusion for molecular conformer generation. Advances in Neural Information Processing Systems, 35:24240–24253, 2022.
- [9] Bowen Jing, Ezra Erives, Peter Pao-Huang, Gabriele Corso, Bonnie Berger, and Tommi Jaakkola. Eigenfold: Generative protein structure prediction with diffusion models. arXiv preprint arXiv:2304.02198, 2023.
- [10] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.
- [11] Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography, 32(5):922–923, 1976.
- [12] DE Koshland Jr. Application of a theory of enzyme specificity to protein synthesis. *Proceedings* of the National Academy of Sciences, 44(2):98–104, 1958.
- [13] Greg Landrum et al. Rdkit: A software suite for cheminformatics, computational chemistry, and predictive modeling. *Greg Landrum*, 8:31, 2013.
- [14] David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Soc., 2017.
- [15] Yeqing Lin and Mohammed AlQuraishi. Generating novel, designable, and diverse protein structures by equivariantly diffusing oriented residue clouds. arXiv preprint arXiv:2301.12485, 2023.
- [16] Zhihai Liu, Minyi Su, Li Han, Jie Liu, Qifan Yang, Yan Li, and Renxiao Wang. Forging the basis for developing protein–ligand interaction scoring functions. *Accounts of chemical research*, 50(2):302–309, 2017.
- [17] Wei Lu, Qifeng Wu, Jixian Zhang, Jiahua Rao, Chengtao Li, and Shuangjia Zheng. Tankbind: Trigonometry-aware neural networks for drug-protein binding structure prediction. Advances in neural information processing systems, 35:7236–7249, 2022.
- [18] Sean P Meyn and Richard L Tweedie. *Markov chains and stochastic stability*. Springer Science & Business Media, 2012.

- [19] Shuya Nakata, Yoshiharu Mori, and Shigenori Tanaka. End-to-end protein–ligand complex structure generation with diffusion-based generative models. *BMC bioinformatics*, 24(1):1–18, 2023.
- [20] Jerod Parsons, J Bradley Holmes, J Maurice Rojas, Jerry Tsai, and Charlie EM Strauss. Practical conversion from torsion space to cartesian space for in silico protein synthesis. *Journal of computational chemistry*, 26(10):1063–1068, 2005.
- [21] Lauren L Porter and Loren L Looger. Extant fold-switching proteins are widespread. Proceedings of the National Academy of Sciences, 115(23):5968–5973, 2018.
- [22] Zhuoran Qiao, Weili Nie, Arash Vahdat, Thomas F. Miller III au2, and Anima Anandkumar. State-specific protein-ligand complex structure prediction with a multi-scale deep generative model, 2023.
- [23] Tadeo Saldaño, Nahuel Escobedo, Julia Marchetti, Diego Javier Zea, Juan Mac Donagh, Ana Julia Velez Rueda, Eduardo Gonik, Agustina García Melani, Julieta Novomisky Nechcoff, Martín N Salas, et al. Impact of protein conformational diversity on alphafold predictions. *Bioinformatics*, 38(10):2742–2748, 2022.
- [24] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.
- [25] Vignesh Ram Somnath, Matteo Pariset, Ya-Ping Hsieh, Maria Rodriguez Martinez, Andreas Krause, and Charlotte Bunne. Aligned diffusion schr\" odinger bridges. *arXiv preprint arXiv:2302.11419*, 2023.
- [26] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv* preprint arXiv:2011.13456, 2020.
- [27] Hannes Stärk, Octavian Ganea, Lagnajit Pattanaik, Regina Barzilay, and Tommi Jaakkola. Equibind: Geometric deep learning for drug binding structure prediction. In *International conference on machine learning*, pages 20503–20521. PMLR, 2022.
- [28] Brian L Trippe, Jason Yim, Doug Tischer, David Baker, Tamara Broderick, Regina Barzilay, and Tommi Jaakkola. Diffusion probabilistic modeling of protein backbones in 3d for the motif-scaffolding problem. arXiv preprint arXiv:2206.04119, 2022.
- [29] 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*, pages 1–3, 2023.
- [30] Felix Wong, Aarti Krishnan, Erica J Zheng, Hannes Stärk, Abigail L Manson, Ashlee M Earl, Tommi Jaakkola, and James J Collins. Benchmarking alphafold-enabled molecular docking predictions for antibiotic discovery. *Molecular Systems Biology*, 18(9):e11081, 2022.
- [31] Kevin Eric Wu, Kevin K Yang, Rianne van den Berg, James Zou, Alex Xijie Lu, and Ava P Amini. Protein structure generation via folding diffusion. 2022.
- [32] Jason Yim, Brian L Trippe, Valentin De Bortoli, Emile Mathieu, Arnaud Doucet, Regina Barzilay, and Tommi Jaakkola. Se (3) diffusion model with application to protein backbone generation. *arXiv preprint arXiv:2302.02277*, 2023.
- [33] Yangtian Zhan, Zuobai Zhang, Bozitao Zhong, Sanchit Misra, and Jian Tang. Diffpack: A torsional diffusion model for autoregressive protein side-chain packing. *arXiv preprint arXiv:2306.01794*, 2023.
- [34] Yang Zhang and Jeffrey Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, 57(4):702–710, 2004.

Supplementary Material

A Additional methodological details of FlexiDock

A.1 Ligand-conditioned receptor (LCR) score model for apo structure optimization

The LCR score model predicts the score of the internal angles on the backbone of a receptor $\theta \in SO(2)^{n \times 6} = s_{\phi_1}(r; l)$, where *n* is the number of amino acid residues in the receptor. We adopted the same set of six angles along the protein backbone from FoldingDiff [31], including three dihedral torsion angles and three bond angles for each amino acid residues. These six angles dictate the backbone conformation of the protein and can be converted to 3D Cartesian coordinates by the Natural Extension Reference Frame (NeRF) algorithm [20]. In contrast to the torsional updates in the ligand's torsional diffusion process for rotatable bonds, which applies vector rotations in the Cartesian space directly [8], our method updates the angles and uses NeRF to reconstruct the protein backbone from updated angles, producing exact Cartesian coordinates (Sec. D). Our method is suitable for modeling the conformational changes for macro-molecules like proteins.

The LCR model is implemented as an SE(3)-invariant tensor product network with similar architectures to DiffDock's models [5] and operates on heterogeneous geometric graphs constructed from the structures of protein-ligand complexes. The nodes in the graph represent ligand (heavy) atoms, receptor residues (using the position of $C\alpha$ atoms), and receptor backbone atoms (N, $C\alpha$, and C). The edges among the nodes are constructed using the same criteria used in DiffDock.

Compared to the ligand score model, the LCR model operates on a multiscale representation of the protein-ligand complex, where the receptor are composed of backbone atoms N, $C\alpha$, and C rather than $C\alpha$ alone. Additionally, instead of pooling the entire geometric heterogenous graph, the scalar outputs corresponding to the receptor's internal angles are produced by mean-pooling the scalar representations of the receptor residue nodes, followed by a fully connected layer. This architectural design enables the LCR model to perform node-level regression task on the receptor graph to predict the internal angles on the amino acid residues. It's worth noting that both the LCR and the ligand score models contain cross ligand-receptor convolution layers to explicitly learn from inter-molecular interactions when making predictions on one of the two molecules.

Algorithm 1 Inference procedure for LCR, which returns a sample from P(R|L = l, R = r)

1: Inputs: Score predictor $s_{\phi_1}(\boldsymbol{r}; \boldsymbol{l}, t)$, Ligand conformation \boldsymbol{l} , Initial Receptor conformation \boldsymbol{r}_N 2: Outputs: Sampled receptor \boldsymbol{r}_0 3: for n = N to 1 do 4: $t = n/N, \Delta\sigma^2 = t\sigma^2$ 5: Predict score $\alpha = s_{\phi_1}(\boldsymbol{r}_n; \boldsymbol{l}, t)$ 6: $\mathbf{z} \sim \mathcal{N}(0, \Delta\sigma^2 \boldsymbol{I})$ 7: $\Delta\theta \leftarrow (\alpha\Delta\sigma^2 + \mathbf{z}) \mod 2\pi$ 8: $\boldsymbol{r}_{n-1} \leftarrow A(\Delta\theta, \boldsymbol{r}_n)$, where $A : SO(2)^{n \times 6} \times \Omega_R \rightarrow \Omega_R$ 9: end for return Sampled receptor \boldsymbol{r}_0

A.2 Alternating inference and composition mechanism of two diffusion processes

We propose to compose two diffusion processes during the inference time, which we term as alternating inference. Alternating inference emulates the induced-fit theory [12], which describes the molecular docking process in apo setting (Fig. 2b) as a two-step process: the ligand first finds its binding region on the protein receptor with the receptor conformation fixed, then it induces the protein receptor to change its conformation to fit the ligand. Alternating inference allows the above two steps to be repeated alternatively (a Gibbs sampling type procedure) for K > 0 steps or until convergence is achieved i.e., wherein the RMSD between the conformations at steps k and k + 1 are within a certain threshold.

More formally, given input ligand and receptor conformations (l_0, r_0) , $k \in \{1, \ldots, K\}$ the score functions $s_{\phi_1}(r; l)$ and $s_{\phi_2}(l; r)$, we iteratively denoise using the two diffusion models (which model the conditional distributions P(L|R = r), P(R|L = l)) to obtain new ligand and receptor

conformations (l_k, r_k) . Details of the iterative denoising procedure are provided in algorithm 2. In what follows, under certain assumptions, we argue and establish that such an alternating procedure of composing two independently trained diffusion models yields samples (l_K, r_K) from the true joint distribution P(L, R) asymptotically. We relegate the reasoning behind why our assumptions are valid and the proofs to the appendix. The assumptions and theorem statements are also stated more formally in Appendix B.

Assumptions 1 (Diffusion Model Assumptions). (a) The Markov Chains $MC_{L|R}$ and $MC_{R|L}$ induced by the diffusion models P(L|R = r), P(R|L = l) are both irreducible and aperiodic.

- (b) Existence of Stationary distributions for the Markov chains: Markov chains when run individually, converge to their respective stationary distributions P(L|R = r) and P(R|L = l) respectively.
- (c) The score functions are bounded for all inputs $|s_{\phi_1}(r; l)|, |s_{\phi_2}(l; r)| < \infty \ \forall r \in \Omega_R, l \in \Omega_L$. Also, the score functions are smooth with bounded gradients $\forall r \in \Omega_R, l \in \Omega_L$.
- (d) The noise injected and removed via both the Markov Chains have a consistent noise profile such that the step sizes $\eta_1, \ll c_1, \eta_2 \ll c_2$. This is to ensure bounded transition dynamics during the inference procedure.

Assumptions 1(c) and (d) are only (approximately) required in practice to ensure that the diffusion model outputs do not behave in an arbitrary fashion - and to ensure convergence to the respective stationary distributions.

Theorem 1 (Convergence of Alternating Inference). For any given initialization of $\mathbf{r}_0 \in \Omega_R$, $\mathbf{l}_0 \in \Omega_L$ and given Assumptions 1, running the alternating process as given in Algorithm 2 will produce a sequence of states $(\mathbf{l}_k, \mathbf{r}_k), k \in \{1, ..., K\}$ that converges to the joint distribution P(L, R) asymptotically.

The immediate consequence of the theorem is that we can use two independently trained diffusion models in an alternating fashion as an alternative to learning more complex diffusion models which jointly captures receptor and ligand flexibility without sacrificing on correctness.

Algorithm 2 Alternating Inference for Compositional Diffusion Models

- Inputs: Diffusion models which independently model P(L|R = r) and P(R|L = l), Number of alternating rounds = K, initial conformations r₀ ∈ Ω_R, l₀ ∈ Ω_L
- 2: **Outputs:** Samples from the joint distribution P(L, R)
- 3: **Procedure:**
- 4: for k = 1 to K:
- 5: Sample $l_k \sim P(L|R = r_{k-1})$ using DiffDock
- 6: Sample $r_k \sim P(R|L = l_k)$ using LCR (Algorithm 1)
- 7: return $(\mathbf{r}_K, \mathbf{l}_K) \sim P(L, R)$

A.3 Time prediction model and truncated reverse diffusion process

During inference, a critical distinction between the reverse diffusion processes of ligand and receptor lies in their initial states: a ligand starts from a random location and conformation, whereas a receptor starts from a fixed location and a well-formed apo structure. The reverse diffusion process for the receptor only need to model the changes from apo to holo conformations, a process more analogous to image denoising than sampling a diffusion model to generating images from noise. Therefore, rolling out the full reverse diffusion process ($t = T \rightarrow t = 0$) in practice may result in over-correction of the input apo structures, leading to conformations far from the desired holo structures.

To combat this issue, we develop a truncated (reverse) diffusion process that initiates the reverse diffusion schedule at $t = T - \hat{t}$ rather than at t = T. \hat{t} is determined by a time prediction model $\hat{t} = g_{\phi_t p}(\mathbf{r}, \mathbf{l})$.

We train this time prediction model using the same noised data from training the LCR model: $\{(l_0, r_t, t), \dots\}$. The time prediction model also share the same architecture with the LCR model, except that it used mean-pooling over the all the receptor residue nodes on the geometric graph to produce a single SE(3)-invariant scalar as the predicted time \hat{t} . We found that the predicted time effectively quantifies the structural differences between an input receptor conformation with its holo counterpart (Fig. S2).

B Statements and proofs

We'll restate the assumptions and theorem statements before the proof for ease of readability.

Let L, R be random variables corresponding to the ligand and receptor which takes values in Ω_L, Ω_R respectively. Let P(L, R) denote the joint distribution of ligands and receptors which form a mutual fit (i.e. holo structure of receptor with correct post of ligand). Our goal is to sample from this joint distribution.

Let the DiffDock and LCR diffusion models, model the conditional probability distributions P(L|R = r), P(R|L = l) which induce the Markov Chains $MC_{L|R}$, $MC_{R|L}$ respectively. Then, if the following assumptions hold,

Assumptions 2 (Diffusion Model Assumptions). (a) The Markov Chains $MC_{L|R}$ and $MC_{R|L}$ induced by the diffusion models P(L|R = r), P(R|L = l) are both irreducible and aperiodic.

- (b) Existence of Stationary distributions for the Markov chains: Markov chains when run individually, converge to their respective stationary distributions π_1, π_2 for P(L|R = r) and P(R|L = l) respectively.
- (c) The score functions are bounded for all inputs $|s_{\theta_1;l}(\mathbf{r})|, |s_{\theta_2;\mathbf{r}}(\mathbf{l})| < \infty \ \forall \mathbf{r} \in \Omega_R, \mathbf{l} \in \Omega_L$. Also, the score functions are smooth with bounded gradients $\forall \mathbf{r} \in \Omega_R, \mathbf{l} \in \Omega_L$.
- (d) The noise injected and removed via both the Markov Chains have a consistent noise profile such that the step sizes $\eta_1, \ll c_1, \eta_2 \ll c_2$. This is to ensure bounded transition dynamics during the inference procedure.

the following statement holds as well:

Statement: Regardless of the initial values for l_0, r_0 , the alternating Gibbs Sampling Procedure which uses the Markov Chains defined by the neural score matching diffusion models will produce sequence of samples l_K, r_K from the joint distribution P(L, R) as $K \to \infty$.

Before, we begin with the proof, adding some preliminaries about Markov Chains:

Irreducibility: A Markov chain is said to be irreducible if there exists a non-zero probability of transitioning (can be more than 1 step to transition) from any state to any other state.

Aperiodicity: A Markov Chain is said to be aperiodic if there are no fixed number of steps at which the chain returns to the starting state for any possible starting state.

Detailed Balance Condition: It is a condition that ensures the existence of a stationary distribution of a Markov chain. For a distribution π to satisfy the condition with respect to a transition matrix P, the following condition must hold for states $a, b: \pi(a)P(a \rightarrow b) = \pi(b)P(b \rightarrow a)$

Proof. We will follow a three step approach to prove the statement. We will first establish the detailed balance condition for the Gibbs Sampling procedure and show that it is satisfied. Subsequently we will look at the irreducibility and aperiodicity of the Gibbs Sampler Markov Chain and then finally the convergence to the stationary distribution.

Lets start by considering two joint states (l_k, r_k) and (l_{k+1}, r_{k+1}) . The associated transition probabilities are given by $P(l_k \rightarrow l_{k+1}|r_k)$ and $P(r_k \rightarrow r_{k+1}|l_{k+1})$. The transition probability from $(l_k, r_k) \rightarrow (l_{k+1}, r_{k+1})$ is given by

$$P((l_k, r_k) \to (l_{k+1}, r_{k+1})) = P(l_k \to l_{k+1} | r_k) P(r_k \to r_{k+1} | l_{k+1})$$

and the reverse transition probability from $(l_{k+1}, r_{k+1}) \rightarrow (l_k, r_k)$ is given by

$$P((l_{k+1}, r_{k+1}) \to (l_k, r_k)) = P(l_{k+1} \to l_k | r_{k+1}) P(r_{k+1} \to r_k | l_k)$$

The detailed balance condition for the Gibbs Sampler is then given by

$$P(l_k, r_k)P(l_k \to l_{k+1}|r_k)P(r_k \to r_{k+1}|l_{k+1}) = P(l_{k+1}, r_{k+1})P(l_{k+1} \to l_k|r_{k+1})P(r_{k+1} \to r_k|l_k)$$

To show that the detailed balance condition holds, we call upon the definitions of conditional probabilities i.e. P(L, R) = P(L|R)P(R) = P(R|L)P(L). Substituting $P(l_k, r_k) = P(l = l_k|R = r_k)P(R = r_k)$ and $P(l_{k+1}r_{k+1} = P(R = r_{k+1}|L = l_{k+1})P(L = l_{k+1})$ in to the equation followed by the definition of the Gibbs Sampler i.e. $P(l_k \to l_{k+1}|r_k) = P(L = l_{k+1}|R = r_k)$ and $P(r_k \to r_{k+1}|L = l_{k+1})$, we see that all the terms on the LHS and the RHS of the equation cancel out.

Therefore, the condition for detailed balance is satisfied for the Markov chain whose states transition from $(l_k, r_k) \rightarrow (l+k+1, r_{k+1})$ and hence implies that P(L, R) is indeed the stationary distribution of the Markov Chain generated by the Gibbs sampling procedure [14].

Now, for any states (l, r) and (l_{k+1}, r_{k+1}) of the Markov chain, given the Gibbs sampling procedure and the properties of the two Markov chains of the conditional distributions, it is clear to see that $P(L = l_{k+1}|R = r_k) > 0$ for any r_k and $P(R = r_{k+1}|L = l_k) > 0$ for any l_k . Therefore our Markov Chain is irreducible (pathway to go from any state to any other state). Also since we are using Gibbs sampling and alternatively sampling from P(L|R) and P(R|L) - theres no fixed loops bound to occur - and our Markov Chain doesn't get trapped into periodic cycles, and therefore our Markov Chain is aperiodic as well.

Now, given that our Gibbs samplers Markov Chain has a stationary distribution and is also aperiodic and irreducible, we use the Fundamental Theorem of Markov Chains [18] to say that we have a guaranteed convergence to P(L, R) regardless of the starting state as $K \to \infty$.

C Additional results

C.1 Rigid-body docking performs poorly in apo setting

We examined the performance of DiffDock, the state-of-the-art rigid-body docking method, on both holo and apo settings. Consistent with [5], we also observed a drastic performance drop in terms of predicting the correct ligand conformations. The percentage of generated ligands conformations below 2Å decrease from 54.45% to 34.33% on the PDBBind dataset (Table S1). We next asked the potential causes of this drop by performing error analysis focusing on the input apo protein conformations. We found a strong correlation (Spearman's ρ =0.298) between the drop in ligand RMSD and the backbone RMSD between ESMFold predicted apo structures and holo structures (Fig. S1). It suggests that rigid-body docking methods such as DiffDock exhibit deteriorated performance when docking ligands to receptors with large conformational discrepancies between their apo and holo states. After all, rigid-body docking methods ignores the conformational flexibilities of the protein receptors. These observations indicate the need to develop a model to explicitly model the proteins' conformational flexibility to help optimize the apo structures. We hypothesize that an apo conformation optimized towards its holo counterpart will improve the docking performance for both the receptor and the ligand.

Table S1: RMSD-based metrics from Apo and Holo settings. The docking performance of DiffDock drop significantly in Apo setting compared to Holo setting, where the perfect apo protein structures are given to the model. All receptors from PDBBind test set (n=268) are used.

	Top-1 L	igand RMSD	Backbone RMSD		
Settings	%<2↑	median ↓	%<2↑	median \downarrow	
Holo	54.48	1.87	100	0	
Apo ESMFold proteins	34.33	4.32	55.97	1.65	



Figure S1: Correlation among backbone and ligand RMSDs from apo and holo settings (on all receptors from PDBBind test set, n=268). Spearman's rho is shown in the figure. Each dot in the scatter plots correspond to a PDB ID (protein-ligand pair). Top-1 pose are selected using DiffDock's confidence model.

C.2 Time model accurately predicts the structural differences between apo and holo conformations

We found that the time prediction model's ability to accurately quantify the structural differences between a pair of apo and holo conformations (Fig S2).



Figure S2: Time prediction model accurately predict the differences between apo and holo structures. Scatter plots show the predicted time on y-axis and structural differences measurements on x-axis (RMSD, TM-score, and pLDDT) for protein complexes in the PDBBind test set. The structural differences are calculated between ESMFold-predicted apo structures and their holo structures.

C.3 Qualitative case studies

We examine a few FlexiDock-generated holo structures and contrasting them with their apo counterparts. Figure S3 shows three examples from the PDBBind test set. As demonstrated by the close TM-score, the predicted backbone structures from our model shares very similar folds with the input apo structures. However, we found our model is able to modify the backbone angles around the coiled regions to move an entire alpha-beta domain in AKT1 closer to its holo conformations (Fig S3a, S3b). We also noticed that it can significantly lower the backbone RMSD of the viral protein VP1 by modifying the coiled regions (Fig. S3c). Similarly, we observed our model is able to bend the coiled regions of input apo conformations while keeping the alpha-helixes and beta-sheets intact (Fig. S4). These observations suggest that our model has learned to retain the relative stable secondary structures in a protein and modify the coiled regions towards a more favorable conformation upon ligand binding. We also noted that the improvement of local structural alignment metrics such as RMSD are independent of the ligand-binding regions, giving our method the advantage of modeling large domain movements involved in apo/holo conformational switches. However, more investigations are needs to enable modeling of fold-switching proteins [4], where certain regions of the protein can assume distinct stable secondary and tertiary structures [21].



Figure S3: **Examples of predicted protein backbones from the PDBBind test set.** Holo, apo, and predicted receptor protein backbone structures are colored in cyan, magenta, and yellow, respectively. (a) AKT1 in Complex with Covalent-Allosteric AKT Inhibitor 15c (b) AKT1 in Complex with Covalent-Allosteric AKT Inhibitor 27 (c) deaminated P domain from norovirus strain Saga GII-4 in complex with Fuc



Figure S4: **Examples of predicted protein backbones from the Saldano dataset.** Holo, apo, and predicted receptor protein backbone structures are colored in cyan, magenta, and yellow, respectively. (a) nudix enzyme AP4A hydrolase in complex with ATP (b) PCAF bromodomain with small chemical ligand NP2 (c) Riboflavin kinase Mj0056 from Methanocaldococcus jannaschii in complex with CDP

D Model training

This section provides additional details on the training of the LCR score model $s_{\phi_1}(\mathbf{r}; \mathbf{l}, t)$ and the time prediction model $g_{\phi_{tp}}(\mathbf{r}; \mathbf{l})$. For the ligand score model $s_{\phi_2}(\mathbf{l}; \mathbf{r}, t)$, we adopted the trained model checkpoint from the DiffDock [5] publication and refer readers to the original publication.

The training procedures are similar for the LCR (Algorithm 3) and the time prediction models (Algorithm 4). The training dataset is composed of holo structures of protein-ligand pairs (r, l) in the same Cartesian coordinate space. We then apply the forward diffusion process to generated noised receptor conformations while keeping the ligand fixed, such that the noised receptor-ligand complex becomes (r_t, l) . To enable the angular noising of the receptor conformation and to ensure the noised receptor r_t sharing the same orientation and position relative to the receptor r, the transformation action for the receptor takes the following form:

$$\boldsymbol{r}_t = A(\Delta\theta, \boldsymbol{r}) = A_{Align}(A_{NeRF}(A_{\theta}(\boldsymbol{r}) + \Delta\theta), \boldsymbol{r})$$

where $A_{\theta}: \Omega_R \to SO(2)^{n \times 6}$ computes the internal angles from the receptor conformation; $A_{NeRF}: SO(2)^{n \times 6} \to \Omega_R$ denotes the Natural Extension Reference Frame (NeRF) algorithm [20] that reconstructs a protein receptor's backbone coordinates from its internal angles. We use the following empirical bond lengths for the NeRF algorithm: $N-C\alpha = 1.46\text{\AA}, C\alpha - C = 1.54\text{\AA}, C-N = 1.34\text{\AA}$. Lastly, $A_{Align}: \mathbb{R}^{n \times 3} \times \mathbb{R}^{n \times 3} \to \mathbb{R}^{n \times 3}$ aligns a pair of Cartesian coordinates by minimizing their RMSD using the Kabsch algorithm [11].

Algorithm 3 Training procedure for LCR score model $s_{\phi_1}(\boldsymbol{r}; \boldsymbol{l}, t)$

- 1: **Inputs:** Training pairs $\{(r, l)\}$
- 2: foreach r, l do
- 3: Sample $t \sim \text{Uni}([0, 1])$
- 4: Sample $\Delta \theta$ from diffusion kernel $p_t(\cdot|0)$
- 5: Compute $\boldsymbol{r}_t \leftarrow A(\Delta \theta, \boldsymbol{r})$
- 6: Predict score $\alpha \in SO(2)^{n \times 6} = s_{\phi_1}(\boldsymbol{r}_t; \boldsymbol{l}, t)$
- 7: Take optimization step on loss $\mathcal{L}(\phi_1) = \|\alpha \nabla \log p_t(\Delta \theta | 0) \|^2$

Algorithm 4 Training procedure for the time prediction model $g_{\phi_{lp}}(\boldsymbol{r}; \boldsymbol{l})$

1: **Inputs:** Training pairs $\{(r, l)\}$

- 2: foreach r, l do
- 3: Sample $t \sim \text{Uni}([0, 1])$
- 4: Sample $\Delta \theta$ from diffusion kernel $p_t(\cdot|0)$
- 5: Compute $\boldsymbol{r}_t \leftarrow A(\Delta \theta, \boldsymbol{r})$
- 6: Predict time $\hat{t} = g_{\phi_{tp}}(\boldsymbol{r}_t; \boldsymbol{l})$
- 7: Take optimization step on loss $\mathcal{L}(\phi_{tp}) = \|\hat{t} t\|^2$

With the noised ligand-receptor complexes $\{(r_t, l)\}$, the LCR score model and the time predictor model are trained with squared loss to predict the score of the angular noise and the diffusion time, respectively.

Training details. We use Adam [?] as optimizer for the LCR and time predictor models. Both models used for evaluation use the exponential moving average of the weights during training, and we update the moving average after every optimization step with a decay factor of 0.999. We use a batch size of 16 and learning rate of 1e-4. We train models for up to 200 epochs with an early stopping patience of 5 epochs by monitoring validation loss as the convergence criterion. The LCR model was trained on eight 24GB NVIDIA A10G GPUs and converged at 35 epochs. The time predictor model was trained on a single GPU and converged at 40 epochs.

Hyperparameters. Both LCR and time predictor adopt the SE(3)-invariant tensor product network architecture from DiffDock [5], with detailed hyperparameters shown in Table S2.

Models	Conv. layers	# scalar features	# vector features	# parameters
LCR	5	24	6	4.8M
Time predictor	4	12	3	488k
DiffDock score model	6	48	10	20.3M

Table S2: Hyperparameters of the SE(3)-invariant tensor product networks used in this study.