Unified Sampling and Ranking for Protein Docking with DFMDock

Lee-Shin Chu   Sudeep Sarma   Jeffrey J. Gray
Department of Chemical and Biomolecular Engineering
Johns Hopkins University, Baltimore, MD 21218, USA.
Correspondence to jgray@jhu.edu

Abstract

Diffusion models have shown promise in addressing the protein docking problem. Traditionally, these models are used solely for sampling docked poses, with a separate confidence model for ranking. We introduce DFMDock (Denoising Force Matching Dock), a diffusion model that unifies sampling and ranking within a single framework. DFMDock features two output heads: one for predicting forces and the other for predicting energies. The forces are trained using a denoising force matching objective, while the energy gradients are trained to align with the forces. This design enables our model to sample using the predicted forces and rank poses using the predicted energies, thereby eliminating the need for an additional confidence model. Our approach outperforms the previous diffusion model for protein docking, DiffDock-PP, with a sampling success rate of 44% compared to its 8%, and a Top-1 ranking success rate of 16% compared to 0% on the Docking Benchmark 5.5 test set. In successful decoy cases, the DFMDock Energy forms a binding funnel similar to the physics-based Rosetta Energy, suggesting that DFMDock can capture the underlying energy landscape.

1 Introduction

1.1 Classical docking methods

Protein-protein docking predicts the structure of a protein complex from the structures of its individual unbound partners [1]. Classical methods involve two key components: (1) sampling algorithms, such as exhaustive global searches, local shape-matching, and Monte Carlo algorithms, generate possible docked structures, while (2) scoring functions evaluate these structures based on physical energy, structural compatibility, or empirical data [2]. An alternative approach is template-based docking, which leverages sequence similarity and evolutionary conservation [3]. However, these methods are time-consuming due to the extensive search and evaluation processes involved. Hence, this work aims to develop a fast and accurate deep-learning method for sampling and scoring protein complexes.

1.2 Related work

Co-folding models. Co-folding models, which predict protein complex structure from sequences, have emerged as powerful tools for addressing protein docking. By leveraging large datasets of protein sequences [4] and structures [5], these models have shown remarkable accuracy in predicting protein complex structures. AlphaFold2 [6] and RoseTTAFold [7] marked a significant breakthrough in protein structure prediction. Originally developed for monomer structure prediction, extensions for multimers were released soon after, making these models the preferred approach for most protein complex predictions. However, challenges remain, such as the time-consuming multiple sequence alignment (MSA) searches and lower accuracy in predicting antibody-antigen interactions

[8]. Recently, AlphaFold3 [9] introduced a diffusion module, replacing the previous structure module, and expanded its capabilities to predict interactions not only between proteins but also with DNA, RNA, and small molecules; but AlphaFold3 still fails in 40% of antibody-antigen cases.

Regression-based models. Unlike co-folding models, regression-based models generally input the individual protein structures, either in 3D or as distance matrices, without relying on MSA. EquiDock [10] was the first model to apply equivariant neural networks for rigid protein docking. While its theoretical framework is robust, its success rate is lower compared to traditional and co-folding models. Following a similar approach, ElliDock [11] introduced elliptic-paraboloid interface representations but did not significantly improve performance. In contrast, GeoDock [12] and DockGPT [13] adopted architectures resembling AlphaFold2, using individual protein structures without MSA while allowing flexible backbones. Although they outperform EquiDock, these methods still underperform compared to co-folding approaches. The limitations of regression-based models are (1) they generate only a single prediction per model, and (2) they are less accurate for predicting protein interactions beyond the training data compared to co-folding models with MSAs.

Diffusion models. Diffusion models [14–16] have been applied to protein docking [17]. Unlike regression-based objectives, DiffDock [18] reformulates docking as a generative process, training the model through denoising score matching [19] on the translation, rotation, and torsion spaces of small molecules. DiffDock-PP [20] adapts DiffDock for protein-protein docking and diffuses only along the translation and rotation spaces (rigid docking). DiffMaSIF [21] follows a similar framework but incorporates additional protein interface embeddings. LatentDock [22] applies diffusion in the latent space, first training a variational autoencoder [23] and then diffusing in the encoder's output, akin to Stable Diffusion [24]. All these models comprise a sampling model, which generates diverse poses through reverse diffusion steps, and a confidence model, which ranks these poses based on their confidence scores.

Energy-based models. Energy-based models [25] train neural networks to approximate the underlying energy function of the training data. DockGame [26] introduces a framework that trains energy functions either supervised by physics-based models or self-supervised through denoising score matching. EBMDock [27] employs statistical potential as its energy function and uses Langevin dynamics to sample docking poses. Arts *et al.* [28] developed diffusion model-based force fields for coarse-grained molecular dynamics by parameterizing the energy of an atomic system and training the gradient of the energy to match the denoising force. With this learned force field, they can both sample from the distribution and perform molecular dynamics simulations. Based on the theory that diffusion models learn the underlying training data distribution, which if well approximated can relate to the energy of the generated samples. DSMBind [29] adopts a similar framework for protein-protein interactions, demonstrating that the learned energy function correlates more strongly with binding energy than previous methods.

Building on these approaches, we propose DFMDock, a diffusion generative model for protein docking. To our knowledge, it is the first model to integrate sampling and ranking within a single framework, utilizing forces and energies learned from diffusion models. Our results show that DFMDock outperforms DiffDock-PP in both sampling and Top-1 ranking success rates. Furthermore, the learned energy function exhibits a binding funnel similar to the physics-based Rosetta energy function [30], suggesting that DFMDock can capture the underlying energy landscape for some protein-protein interactions.

2 Methods

2.1 Model

According to statistical physics, a docking pose x is a random state sampled from the Boltzmann distribution: $p(x) = \frac{e^{-E(x)/k_B T}}{Z}$, where $E(x)$ is the energy, k_B is the Boltzmann constant, T is the temperature, and $Z = \int_\Omega e^{-E(x)/k_B T}$ is the partition function. For rigid docking between a "receptor" protein and a "ligand" protein, the search space Ω spans all possible translations and rotations of the ligand, with the receptor fixed. The per-residue forces on the ligand are given by:

$$f_i = -\nabla_{r_i} E(x) = \nabla_{r_i} \log p(x), \tag{1}$$

where r_i are the C_α coordinates of the ligand residues relative to their center. Our goal is to learn f_i and $E(x)$, using f_i for sampling docking poses and $E(x)$ for ranking.

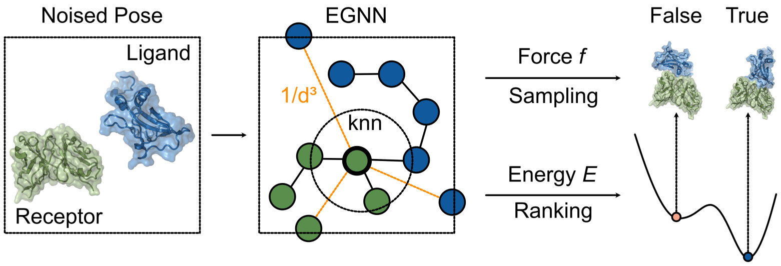


Figure 1: DFMDock model overview.

We use an equivariant graph neural network (EGNN) [31] to predict the energy E and the forces f_i on each residue:

$$h^L, x^L = \text{EGNN}(h^0, x^0, e_{ij}). \tag{2}$$

h represents the node embeddings, which for the input h^0 concatenate the amino acid sequence one-hot encoding with the ESM2 (650M) embeddings [32]. x is the set of C_α coordinates, and e_{ij} is an edge embedding of trRosetta [33] geometry and relative positional encoding [34]. To balance short- and long-range interactions, we construct graphs of 20 nearest neighbor edges and 40 edges selected randomly using an inverse cubic distance weighting [35]. The predicted energy is computed as the average output of a multi-layer perceptron (MLP) ϕ_E that inputs the concatenated node representations of ligand i and receptor j, where the distance between them, d_{ij}, is within a cutoff distance d:

$$E = \frac{1}{N} \sum_{(i,j):d_{ij}<d} \phi_E(h_i^L \| h_j^L), \tag{3}$$

where N is the number of residue pairs within the cutoff distance. The predicted forces on each ligand residue are derived as the displacement vectors between the input coordinates and the updated coordinates:

$$f_i = x_i^L - x_i^0. \tag{4}$$

The translational force for the ligand is obtained by averaging the per-residue forces: $F_{\text{translation}} = \frac{1}{n}\sum_{i=1}^{n} f_i$. In the Appendix, we show that the gradient of energy with respect to a rotation vector ω is given by $\nabla_\omega E = \frac{1}{n}\sum_{i=1}^{n}(r_i \times \nabla_{r_i} E)$. Thus, the rotational force for the ligand is: $F_{\text{rotation}} = -\nabla_\omega E = \frac{1}{n}\sum_{i=1}^{n} r_i \times f_i$.

To improve numerical stability [36], we normalize the translational and rotational forces and use two MLPs ϕ that input the unnormalized magnitude and timestep t, to learn the scaling factors:

$$\hat{F}_{\text{translation}} = \frac{F_{\text{translation}}}{||F_{\text{translation}}||} \cdot \phi_{\text{translation}}(||F_{\text{translation}}||, t), \tag{5}$$

$$\hat{F}_{\text{rotation}} = \frac{F_{\text{rotation}}}{||F_{\text{rotation}}||} \cdot \phi_{\text{rotation}}(||F_{\text{rotation}}||, t). \tag{6}$$

We train the model using denoising force matching for both translation and rotation:

$$L_{\text{translation}} = \left\| \hat{F}_{\text{translation}} - \nabla_{\hat{x}} \log p(\hat{x}|x) \right\|^2, \tag{7}$$

$$L_{\text{rotation}} = \left\| \hat{F}_{\text{rotation}} - \nabla_{\hat{\omega}} \log p(\hat{\omega}|\omega) \right\|^2, \tag{8}$$

where $p(\hat{x}|x)$ denotes the conditional probability distribution of the noised translation $\hat{x}$ given the true translation x, modeled as a Gaussian distribution in $\mathbb{R}^3$, and $p(\hat{\omega}|\omega)$ represents the conditional probability distribution of the noised rotation $\hat{\omega}$ given the true rotation ω, modeled as an isotropic Gaussian distribution on SO(3) [37–39].

To train the energy function for ranking, the energy conservation loss [40] is calculated as the mean squared error between the predicted forces and the negative gradient of the predicted energy with respect to the coordinates:

$$L_{\text{conservation}} = \left\| -\nabla_{r_i} E(x) - f_i \right\|^2. \tag{9}$$

To ensure that the global energy minimum aligns with the ground truth [41, 42], the energy contrastive loss is defined as:

$$L_{\text{contrastive}} = -\log\left(\frac{e^{-E_{\text{gt}}}}{e^{-E_{\text{gt}}} + e^{-E_{\text{noised}}}}\right), \tag{10}$$

where E_{gt} and E_{noised} are the energies of the ground truth and noised structures. The final loss function combines these components:

$$L = L_{\text{translation}} + L_{\text{rotation}} + L_{\text{conservation}} + L_{\text{contrastive}}. \tag{11}$$

2.2 Data

We trained our model on DIPS-hetero, a subset of DIPS [43, 44] with approximately 11k heterodimers. For testing, we evaluated it on 25 targets, as selected in the EquiDock report, from the Docking Benchmark 5.5 (DB5.5) [45], a widely used dataset for assessing docking performance.

3 Results and Discussion

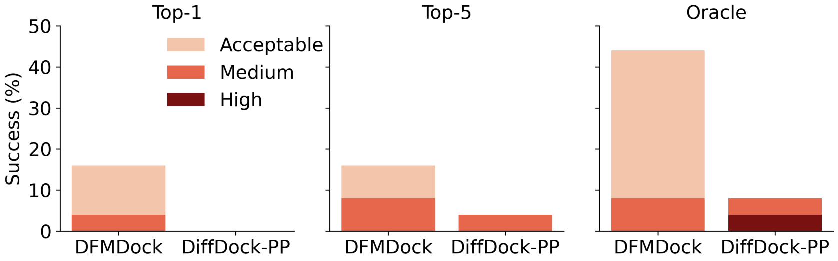


Figure 2: Success rates of DFMDock and DiffDock-PP in Top-1, Top-5, and Oracle settings, categorized by Acceptable (DockQ > 0.23), Medium (DockQ > 0.49), and High (DockQ > 0.80) accuracy ratings [46].

3.1 DFMDock achieves higher success rate than DiffDock-PP

We compared DFMDock to DiffDock-PP (trained on DIPS) on the DB5.5 test set. Both models generated 120 samples per target, each initialized from different starting positions, using 40 diffusion steps. DFMDock samples were ranked using the model's energy function (DFMDock Energy) and DiffDock-PP samples were ranked using it's confidence model. As shown in Figure 2, DFMDock consistently outperforms DiffDock-PP across all settings, with the largest margin observed in the Oracle setting. (Here Oracle refers to the highest DockQ among all samples per target). While DiffDock-PP achieved state-of-the-art success on the DIPS test set, its performance on DB5.5 dropped significantly (8%), likely due to data leakage between the DIPS training and test sets [12, 47]. In contrast, DFMDock demonstrates better generalization to protein-protein interactions in DB5.5.

3.2 DFMDock learns physics-like energy

To evaluate our model's energy function, we plotted both the DFMDock Energy and the Rosetta Energy vs the Interface RMSD and DockQ (two measures of docking accuracy) for the 120 DFMDock-generated samples per target. Figure 3a shows similar binding funnels under both scoring methods for a sample target (PDB ID: 2SIC), suggesting that DFMDock captures the physical energy landscape of protein docking. This funnel-like behavior indicates the model can distinguish between near-native and non-native docking poses, making it valuable for ranking docking decoys. The contour plot of Rosetta Energy vs DFMDock Energy shows that DFMDock ranks medium and acceptable quality predictions higher than incorrect predictions. Figure 3b shows that DFMDock energy slightly outperforms Rosetta in identifying acceptable quality poses (4 vs 3 targets) but underperforms in discriminating medium quality structures (1 vs 2 targets). In 7 out of 11 cases where DFMDock succeeded in sampling, both scoring methods failed to rank the poses correctly, suggesting that the

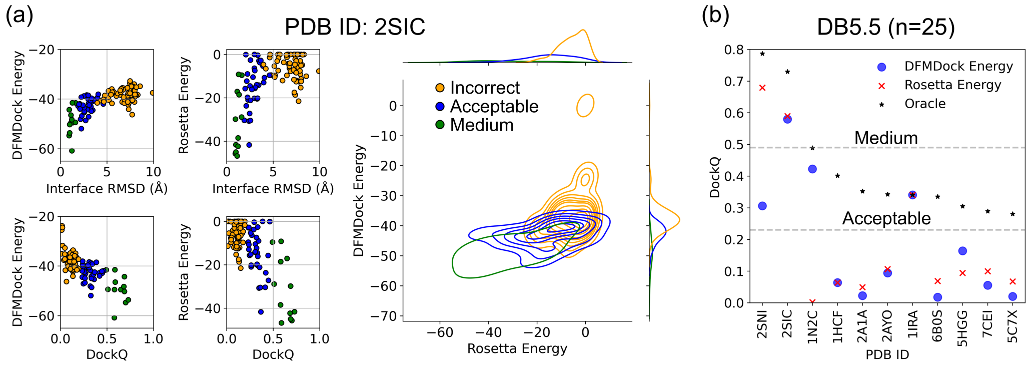
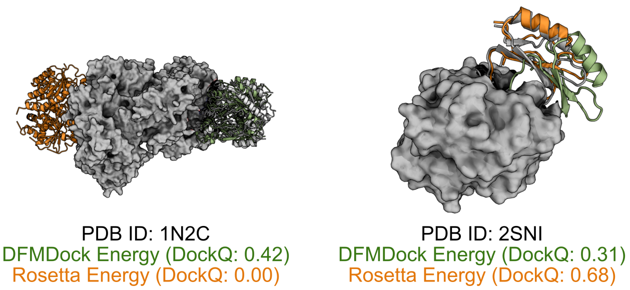


Figure 3: Comparison of ranking methods between DFMDock and Rosetta. (a) Binding funnels (Interface RMSD and DockQ) and contour plot (DFMDock Energy vs Rosetta Energy) for a target from the DB5.5 test set (PDB ID: 2SIC), colored by DockQ ranges: Incorrect (orange), Acceptable (blue), and Medium (green). (b) Comparison of Top-1 ranked DockQ scores for different PDB IDs: DFMDock Energy (blue circles), Rosetta Energy (red crosses), and Oracle (black stars). The dashed line at DockQ=0.23 and DockQ=0.49 indicates the Acceptable and Medium threshold. Only successful decoy sampling cases from DB5.5 (11 out of 25) are shown.

sampled poses may be of lower quality or contain steric clashes, thus requiring further development of the model.

Figure 4 shows a structural comparison between DFMDock and Rosetta's top-ranked predictions for two samples. For PDB ID 1N2C, DFMDock identifies an acceptable quality pose (DockQ=0.42), while Rosetta identifies an incorrect pose (DockQ=0.00). However, for 2SNI, DFMDock fails to identify medium quality structures as effectively as Rosetta energy, suggesting that DFMDock's energy function is less accurate in this case. Incorporating all-atom details into DFMDock's energy function could help address this issue, enhancing its ability to distinguish between acceptable and medium-quality docking poses and improving its overall reliability across diverse protein-protein interactions.

Figure 4: Top-1 predictions ranked by DFMDock Energy (green) and Rosetta Energy (orange) aligned to the ground truth structures (grey).


PDB ID: 1N2C
DFMDock Energy (DockQ: 0.42)
Rosetta Energy (DockQ: 0.00)
PDB ID: 2SNI
DFMDock Energy (DockQ: 0.31)
Rosetta Energy (DockQ: 0.68)

4 Conclusion

DFMDock is a generative diffusion model that integrates sampling and ranking for protein docking. DFMDock outperforms DiffDock-PP on the DB5.5 test set and generates binding funnels comparable to those from the Rosetta interface score, highlighting its ability to mimic physical interactions. However, in many cases where DFMDock succeeded in sampling, it failed to rank poses accurately, indicating the need for further development. Additionally, its accuracy in identifying medium-quality structures can be further optimized. Currently, the model is trained on a limited dataset, which may constrain its generalization. With the availability of larger and more comprehensive datasets [48], future work will incorporate all-atom details to enhance DFMDock's precision and reliability across a broader range of protein interactions.

Acknowledgements

This work was supported by National Institutes of Health grant R35-GM141881 and by Moderna. Computational resources were provided by the Advanced Research Computing at Hopkins (ARCH). The authors thank Jeremias Sulam for valuable discussions and insightful feedback.

Code Availability

The inference code, model weights, and test set are available at `https://github.com/Graylab/DFMDock`.

References

[1] Ilya A Vakser. Protein-protein docking: From interaction to interactome. *Biophysical journal*, 107(8):1785–1793, 2014.

[2] Sheng-You Huang. Search strategies and evaluation in protein–protein docking: principles, advances and challenges. *Drug discovery today*, 19(8):1081–1096, 2014.

[3] Andras Szilagyi and Yang Zhang. Template-based structure modeling of protein–protein interactions. *Current opinion in structural biology*, 24:10–23, 2014.

[4] The UniProt Consortium. Uniprot: the universal protein knowledgebase. *Nucleic acids research*, 46(5):2699–2699, 2018.

[5] Helen M Berman, Tammy Battistuz, Talapady N Bhat, Wolfgang F Bluhm, Philip E Bourne, Kyle Burkhardt, Zukang Feng, Gary L Gilliland, Lisa Iype, Shri Jain, et al. The protein data bank. *Acta Crystallographica Section D: Biological Crystallography*, 58(6):899–907, 2002.

[6] 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.

[7] Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N Kinch, R Dustin Schaeffer, et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021.

[8] Rui Yin, Brandon Y Feng, Amitabh Varshney, and Brian G Pierce. Benchmarking alphafold for protein complex modeling reveals accuracy determinants. *Protein Science*, 31(8):e4379, 2022.

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

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

[11] Ziyang Yu, Wenbing Huang, and Yang Liu. Rigid protein-protein docking via equivariant elliptic-paraboloid interface prediction. *arXiv preprint arXiv:2401.08986*, 2024.

[12] Lee-Shin Chu, Jeffrey A Ruffolo, Ameya Harmalkar, and Jeffrey J Gray. Flexible protein–protein docking with a multitrack iterative transformer. *Protein Science*, 33(2):e4862, 2024.

[13] Matt McPartlon and Jinbo Xu. Deep learning for flexible and site-specific protein docking and design. *BioRxiv*, pages 2023–04, 2023.

[14] 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.

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[16] 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.

[17] Jason Yim, Hannes Stärk, Gabriele Corso, Bowen Jing, Regina Barzilay, and Tommi S Jaakkola. Diffusion models in protein structure and docking. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 14(2):e1711, 2024.

[18] 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.

[19] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

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

[21] Freyr Sverrisson, Mehmet Akdel, Dylan Abramson, Jean Feydy, Alexander Goncearenco, Yusuf Adeshina, Daniel Kovtun, Céline Marquet, Xuejin Zhang, David Baugher, et al. Diffmasif: Surface-based protein-protein docking with diffusion models. In *Machine Learning in Structural Biology workshop at NeurIPS 2023*, 2023.

[22] Matt McPartlon, Céline Marquet, Tomas Geffner, Daniel Kovtun, Alexander Goncearenco, Zachary Carpenter, Luca Naef, Michael Bronstein, and Jinbo Xu. Latentdock: Protein-protein docking with latent diffusion. *MLSB*, 2023.

[23] Diederik P Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

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

[25] Yilun Du and Igor Mordatch. Implicit generation and modeling with energy based models. *Advances in Neural Information Processing Systems*, 32, 2019.

[26] Vignesh Ram Somnath, Pier Giuseppe Sessa, Maria Rodriguez Martinez, and Andreas Krause. Dockgame: Cooperative games for multimeric rigid protein docking. *arXiv preprint arXiv:2310.06177*, 2023.

[27] Huaijin Wu, Wei Liu, Yatao Bian, Jiaxiang Wu, Nianzu Yang, and Junchi Yan. Ebmdock: Neural probabilistic protein-protein docking via a differentiable energy model. In *The Twelfth International Conference on Learning Representations*, 2024.

[28] Marloes Arts, Victor Garcia Satorras, Chin-Wei Huang, Daniel Zugner, Marco Federici, Cecilia Clementi, Frank Noé, Robert Pinsler, and Rianne van den Berg. Two for one: Diffusion models and force fields for coarse-grained molecular dynamics. *Journal of Chemical Theory and Computation*, 19(18):6151–6159, 2023.

[29] Wengong Jin, Xun Chen, Amrita Vetticaden, Siranush Sarzikova, Raktima Raychowdhury, Caroline Uhler, and Nir Hacohen. Dsmbind: Se (3) denoising score matching for unsupervised binding energy prediction and nanobody design. *bioRxiv*, pages 2023–12, 2023.

[30] Rebecca F Alford, Andrew Leaver-Fay, Jeliazko R Jeliazkov, Matthew J O'Meara, Frank P DiMaio, Hahnbeom Park, Maxim V Shapovalov, P Douglas Renfrew, Vikram K Mulligan, Kalli Kappel, et al. The rosetta all-atom energy function for macromolecular modeling and design. *Journal of chemical theory and computation*, 13(6):3031–3048, 2017.

[31] Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International conference on machine learning*, pages 9323–9332. PMLR, 2021.

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

[33] Jianyi Yang, Ivan Anishchenko, Hahnbeom Park, Zhenling Peng, Sergey Ovchinnikov, and David Baker. Improved protein structure prediction using predicted interresidue orientations. *Proceedings of the National Academy of Sciences*, 117(3):1496–1503, 2020.

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

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

[36] Matthew Masters, Amr Mahmoud, and Markus Lill. Fusiondock: Physics-informed diffusion model for molecular docking. In *ICML2023 CompBio Workshop*, 2023.

[37] Adam Leach, Sebastian M Schmon, Matteo T Degiacomi, and Chris G Willcocks. Denoising diffusion probabilistic models on so (3) for rotational alignment. *ICLR2022 GTRL Workshop*, 2022.

[38] 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.

[39] Yesukhei Jagvaral, Francois Lanusse, and Rachel Mandelbaum. Unified framework for diffusion generative models in so (3): applications in computer vision and astrophysics. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

[40] Alexandre Agm Duval, Victor Schmidt, Alex Hernández-Garcıa, Santiago Miret, Fragkiskos D Malliaros, Yoshua Bengio, and David Rolnick. Faenet: Frame averaging equivariant gnn for materials modeling. In *International Conference on Machine Learning*, pages 9013–9033. PMLR, 2023.

[41] Yilun Du, Jiayuan Mao, and Joshua B Tenenbaum. Learning iterative reasoning through energy diffusion. *arXiv preprint arXiv:2406.11179*, 2024.

[42] Changsoo Lee, Jonghun Won, Seongok Ryu, Jinsol Yang, Nuri Jung, Hahnbeom Park, and Chaok Seok. Galaxydock-dl: Protein–ligand docking by global optimization and neural network energy. *Journal of Chemical Theory and Computation*, 2024.

[43] RJL Townshend, R Bedi, PA Suriana, and RO Dror. End-to-end learning on 3d protein structure for interface prediction. arxiv. *arXiv:1807.01297*, 2018.

[44] Alex Morehead, Chen Chen, Ada Sedova, and Jianlin Cheng. Dips-plus: The enhanced database of interacting protein structures for interface prediction. *Scientific data*, 10(1):509, 2023.

[45] Thom Vreven, Iain H Moal, Anna Vangone, Brian G Pierce, Panagiotis L Kastritis, Mieczyslaw Torchala, Raphael Chaleil, Brian Jiménez-García, Paul A Bates, Juan Fernandez-Recio, et al. Updates to the integrated protein–protein interaction benchmarks: docking benchmark version 5 and affinity benchmark version 2. *Journal of molecular biology*, 427(19):3031–3041, 2015.

[46] Sankar Basu and Björn Wallner. Dockq: a quality measure for protein-protein docking models. *PloS one*, 11(8):e0161879, 2016.

[47] Anton Bushuiev, Roman Bushuiev, Jiri Sedlar, Tomas Pluskal, Jiri Damborsky, Stanislav Mazurenko, and Josef Sivic. Revealing data leakage in protein interaction benchmarks. *arXiv preprint arXiv:2404.10457*, 2024.

[48] Daniel Kovtun, Mehmet Akdel, Alexander Goncearenco, Guoqing Zhou, Graham Holt, David Baugher, Dejun Lin, Yusuf Adeshina, Thomas Castiglione, Xiaoyun Wang, et al. Pinder: The protein interaction dataset and evaluation resource. *bioRxiv*, pages 2024–07, 2024.

Appendix

Gradient of energy E with respect to a rotation vector ω

Consider a rigid body with N points labeled $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N$, where each $\mathbf{r}_i = \mathbf{x}_i - \mathbf{c}$ represents the position of point i relative to the center of mass $\mathbf{c}$. When the rigid body undergoes a small rotation $d\omega$, where ω is a rotation vector, the displacement $d\mathbf{r}_i$ of each point $\mathbf{r}_i$ is given by:

$$d\mathbf{r}_i = d\omega \times \mathbf{r}_i$$

The energy E of the system depends on the positions $\mathbf{r}_i$. The change in energy dE due to small displacements $d\mathbf{r}_i$ of the points is:

$$dE = \sum_{i=1}^{N} \nabla_{\mathbf{r}_i} E \cdot d\mathbf{r}_i$$

where $\nabla_{\mathbf{r}_i} E$ is the gradient of the energy with respect to the position of point $\mathbf{r}_i$. (In this work we assume the direct energy dependence on rotation angle of each residue is small.)

Substituting the expression for the displacement $d\mathbf{r}_i$, we get:

$$dE = \sum_{i=1}^{N} \nabla_{\mathbf{r}_i} E \cdot (d\omega \times \mathbf{r}_i)$$

Using the scalar triple product identity:

$$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c}) = \mathbf{b} \cdot (\mathbf{c} \times \mathbf{a})$$

we can simplify the expression for dE:

$$dE = \sum_{i=1}^{N} d\omega \cdot (\mathbf{r}_i \times \nabla_{\mathbf{r}_i} E)$$

Thus, the gradient of energy with respect to the rotation vector ω is:

$$\nabla_\omega E = \sum_{i=1}^{N} (\mathbf{r}_i \times \nabla_{\mathbf{r}_i} E)$$

where $\mathbf{r}_i$ is the vector from the center of mass to the point $\mathbf{x}_i$, and $\nabla_{\mathbf{r}_i} E$ is the gradient of the energy with respect to the position of point $\mathbf{r}_i$.

If we have a finite number of points N, we can normalize the sum by the number of points to get an average gradient:

$$\nabla_\omega E = \frac{1}{N} \sum_{i=1}^{N} (\mathbf{r}_i \times \nabla_{\mathbf{r}_i} E)$$

This provides an averaged estimate of the gradient of energy with respect to the rotation vector ω when dealing with a finite set of points on the rigid body.