Using Flexible Docking and Reinforcement Learning to Guide Antibody Design

Ananth Shyamal, Mohamed Samb, Wilson Ho

Abstract

Antibody-antigen interactions are a key component of immune responses, helping identify pathogens and defend the body against them. Furthermore, antibodies are extensively studied as therapeutic modalities for various diseases, such as cancer. Modeling these interactions is of significant clinical relevance, but it is very experimentally costly and time-consuming. These problems make *in silico* models for designing antibodies and predicting antigen-antibody docking especially useful. High-affinity binder design can be driven by many different approaches. Existing docking models are often able to perform simultaneous docking and sequence-structure co-design. HERN (Jin et al., 2022) generates an antibody sequence autoregressively, iteratively predicting the next residue and performing one-step structure refinement. By mimicking directed evolution, reinforcement learning also has the ability to guide *in silico* protein design by iterating through rounds of random mutation and candidate scoring. EvoPlay (Wang et al., 2023) demonstrates this, using reinforcement learning in its implementation of a neural network-guided Monte Carlo tree search. EvoPlay's reward function uses residue distance and contact features, which fail to provide a complete description of peptide binding interactions. In this paper, we propose redesigning EvoPlay to support HERN as a reward model and incorporate interaction energies into the reward function. Starting with native antibodies selected from the SAbDab (Dunbar et al., 2014) database, we used the combined model to mutate residues in the CDR-H3 paratope of the heavy chain. The computed energy of the resulting structures indicates higher binding affinity than structures generated by HERN alone. In many cases, the resulting paratope also has a lower interaction energy than the native structure. Further study is needed to evaluate the quality of these paratopes, such as their stability when incorporated in the full antibody structure. By incorporating interaction energies into EvoPlay's reward function, we are able to produce high-affinity antibody sequences, which could have significant applications in fields such as immunology.

Background

Antibodies are proteins produced by the immune system and are involved in detecting antigens, which are molecules that stimulate an immune response. Antibodies are composed of two identical heavy and light chains in a Y-shaped configuration. It is estimated that there are up to 100 million unique antibodies within the human body; while the stem of antibodies remains relatively conserved in composition (hence why it is termed the constant domain), the tips of the antibodies are highly variable and function to recognize antigens specifically.

Antibodies and antigens are central players in the body's immune system. Antibodies are produced by B cells and enter circulation to specifically bind to antigens throughout the body. Furthermore, antibodies are able to identify pathogens by, for example, binding to foreign antigens presented on the surface of pathogens. Once the pathogens are tagged by antibodies, they are engulfed and eliminated by phagocytic cells such as macrophages.

Therefore, modeling antigen-antibody interactions is clinically relevant due to the various therapeutic applications of antigens and antibodies, such as vaccine development and monoclonal antibodies. It is experimentally time-consuming and expensive to model these interactions using traditional techniques like X-ray crystallography, NMR spectroscopy, and cryo-EM. Therefore, *in silico* models that predict antigen-antibody docking are particularly useful, and such models may be used to design antibodies for specific epitopes.

Current models for protein docking employ a variety of methods. Physics-based models estimate the potential energy of configurations, which can be expressed as a function of various energy terms, including those associated with forces such as electrostatic and van der Waals. These kinds of models tend to be computationally expensive, which has led to the development of machine-learning-based models that are trained on large sequence and 3D structure datasets. Existing models employ either a rigid or flexible docking scheme. In rigid docking, both the protein and the ligand are modeled as rigid structures. While this paradigm increases computational efficiency, it isn't a realistic model since proteins and molecules change in conformation in biological settings. Additionally, protein mutations can have a much larger effect on side chain conformations and leave the backbone structure relatively unchained; this phenomenon is not effectively captured by rigid docking. Lastly, while docked proteins exhibit shape complementarity, the corresponding unbound protein structures may not. As a result, a rigid docking model (and its training dataset) would capture a small fraction of possible docking configurations. These challenges with rigid docking have led to the development of flexible docking models that consider both backbone and side-chain flexibility when docking proteins.

Several flexible docking deep-learning models have been adapted to design antibodies and predict antibody-antigen docking. In particular, some models focus on predicting binding between a specific region of an antigen (epitope) and a region of an antibody (paratope). One of the state-of-the-art models for this prediction task is HERN (Jin et al., 2022). Given the 3D structure of an antigen, HERN is able to simultaneously fold and dock a paratope (from its sequence) to an epitope on the antigen. Prior methods for antibody-antigen docking were only able to model the protein backbone structure of the complex. HERN introduces a hierarchical equivariance paradigm that enables it to model side chains. In particular, HERN utilizes a hierarchical graph representation of the binding interface by maintaining two graph representations: one for just C_α atoms and one also including side chain atoms. This enables the model to refine side chain configurations while globally updating backbones locally. HERN, like some previous models, maintains equivariance by predicting the force between atoms instead of their 3D locations. Overall, HERN showed significantly improved performance (45% increase in performance on docking baselines) and speed on antibody-antigen docking compared to previous models.

High-affinity binder design can be driven by many different approaches. Existing docking models are often able to perform simultaneous docking and sequence-structure co-design. Transformer-based protein models mask and predict linear segments of a protein chain, analogous to word prediction in a language model. Jin et al. (2022) generate an antibody sequence autoregressively, iteratively predicting the next residue and performing one-step structure refinement.

Inspired by directed evolution, in silico protein design can also be directed by reinforcement learning, using iterative rounds of random mutation and candidate scoring. EvoPlay (Wang et al., 2023) implements a neural network-guided Monte Carlo tree search (MCTS) based on deep learning models for chess games. The optimization process starts from a complete protein sequence and performs a series of moves representing single-site mutations. To evaluate mutants during the tree search process, EvoPlay queries a surrogate model to retrieve an environmental reward. For instance, the surrogate can be a

machine-learning model that maps sequences to functions. The model also supports AlphaFold2 (AF2) (Evans et al., 2022), which allows the incorporation of rewards based on the protein-peptide interface.

EvoPlay's reward function uses residue distance and contact features predicted by AF2. Although these are easy to obtain from a given structure, such features provide a limited description of the interactions involved in peptide binding. Other docking models have experimented with physics-based scoring functions to evaluate protein complexes. Esquivel-Rodríguez et al. (2012), for example, consider a combination of van der Waals, electrostatic, hydrogen bonding, and solvation terms. Compared to other scoring schemes, the rigorous physics-based score provides a higher prediction accuracy. However, the score is considerably more expensive to compute, which makes it a potential bottleneck for model training.

Results and Discussion

Using the same reinforcement learning framework as EvoPlay, we incorporated HERN as an alternative reward model. Since HERN is fine-tuned on flexible complexes, its predicted structures can be used to specifically design better antibodies. We also extended EvoPlay's structure-based reward function by incorporating estimates of binding energy.

Dataset

Since we are using HERN as our reward model, we used the same dataset compiled by Jin et al. (2022). All antibody-antigen complexes came from the structural antibody database (SAbDab) (Dunbar et al., 2014). Of these, 2777 and 169 structures were used by HERN for training and validation, respectively. The remaining 58 structures used for testing cover diverse antigen types. For our model, we randomly selected 10 structures out of the 58 available to test due to the speed of our model and the time constraints of the project. As with Jin et al. (2022), we chose to focus on the heavy chain's CDR-H3 paratope, which is the major determinant of binding affinity.

Baselines

Given a paratope sequence and an antigen structure with a specified epitope, HERN's docking module predicts the 3D coordinates of each paratope atom. The model simultaneously folds and docks the paratope sequence, as the original 3D structure is not given. We used the pre-trained docking model provided in the corresponding GitHub repository. OpenMM relaxation was necessary to refine the final structures, as without the refinement, sidechain clashes within the predicted structure led to extremely high interaction energies.

We used the FoldX library (Delgado et al., 2019) to computationally approximate the interaction energies between two proteins. The FoldX force field takes into account van der Waals forces, solvent approximations, hydrogen bonding, electrostatic contributions, and entropic costs of folding. Given a relaxed structure, the energy calculation is very efficient, running in about a second per structure. Favorable structures have lower interaction energies. However, with docked structures, the computed energy can only be treated as an approximation of the true binding energy.

Figure 1 depicts a docked structure, using PDB entry 4FFV as an example. 4FFV is part of the test set and corresponds to a peptidase complexed with an inhibitory antibody. The epitope was determined from

the original PDB using the antigen residues closest to the native paratope. After OpenMM refinement, the final structure has an interaction energy of 3.55 kcal/mol.

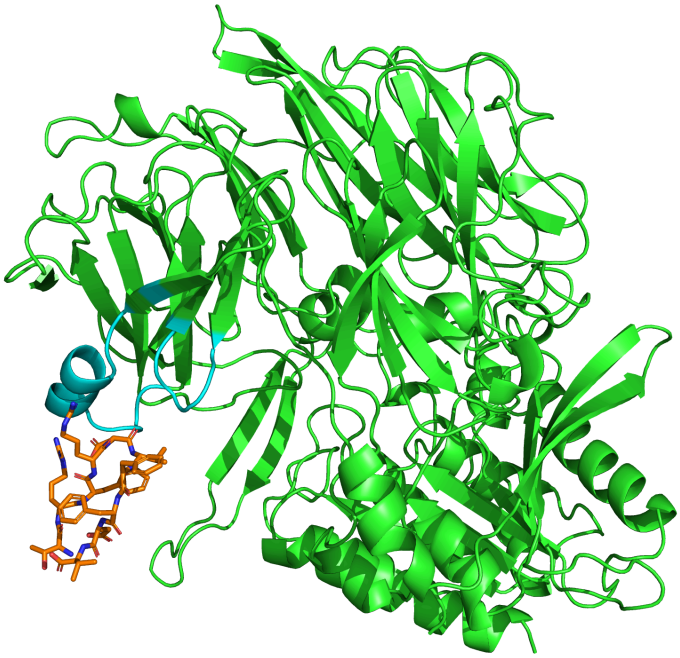

Figure 1: Paratope docking example. PDB 4FFV. Antigen in green. Epitope residues in cyan. Paratope residues in orange.

We ran HERN's autoregressive antibody design module as a baseline to compare against. HERN's decoder runs the docking module to update the generated complex in each iteration. The complex is encoded by another module and used to predict each residue's amino acid type.

Generated sequences are given in the appendix. HERN generates amino acids that are very different from the native sequence but quite similar to one another. The resulting sequences include many tyrosine residues, usually placed one after the other. With its aromatic ring and hydroxyl group, tyrosine is frequently involved in different protein interactions. Glycine is also frequently used. The small amino acid may be responsible for contributing flexibility to the paratope structure.

Figure 2 compares the interaction energies of the native test structures against those generated by HERN. Of the 58 test structures, 8 showed improved interaction energy. Due to time constraints, only one CDR-H3 sequence was sampled for each epitope. Sampling more sequences may result in structures with even lower interaction energies.

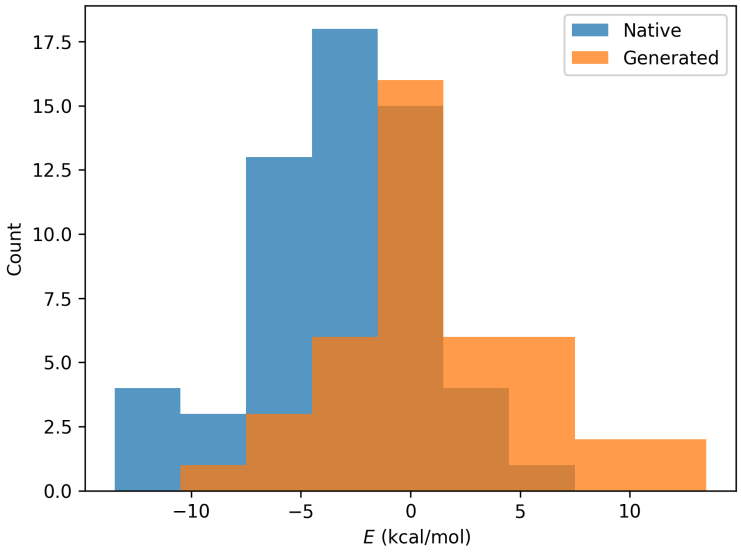

Figure 2: Computed interaction energies. All 58 test sequences. 16 structures with generated $E > 20$ excluded.

Unfortunately, similar design baselines could not be generated with EvoPlay's AF2-based model, as its GitHub repository is missing code for preprocessing input PDBs.

Incorporating HERN in EvoPlay's Reward Model

The EvoPlay model was trained with the existing reinforcement learning framework, iteratively mutating antibody CDR sequences to achieve the best reward. The overall action space consists of 20 possible mutations for each position in the sequence. To characterize the action space, EvoPlay performs a series of MCTS simulations. A simulation consists of walking from an initial root state to a leaf node, where nodes of the tree represent different sequences. The process is driven by a policy-value neural network, which is trained on the environmental reward function so that, generally, nodes with higher rewards are visited more often. Once the search reaches a leaf node, it adds child nodes to the leaf and traces back up the tree, updating edge action values. Once the search returns to the root, a final action is sampled based on visit counts. The residue indicated by the action is mutated, and the state is updated. An episode runs until a mutation results in a state with a lower reward than the previous state, with the policy-value function being updated at the end of each episode.

At each leaf node, the simulation calls HERN and FoldX on the predicted sequence to generate a reward. The reward function was defined using the negative of the interaction energy. Since EvoPlay's policy-value network expects rewards between 0 and 1, we took the sigmoid of the resulting value. Most structures have energies between -5 and 5 kcal/mol, and the sigmoid function provides enough granularity to distinguish the sequences. Some parameter tuning in EvoPlay was necessary in order to efficiently generate structure predictions. We chose to run 10 simulations for each action. We also chose to run the model for five episodes, where each initial state is taken from the best sequence in the previous episode. The best sequence across all episodes was taken as the final output. Figure 3 illustrates the distribution of sequences and interaction energies sampled during the training process for an example test PDB.

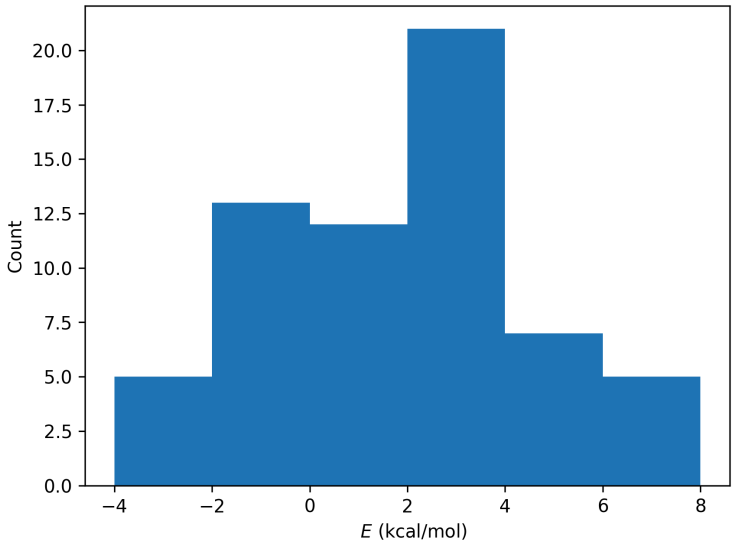

Figure 3: Structures tested during EvoPlay simulation. PDB 3UZQ. 7 mutated structures with $E > 20$ excluded.

Antibody Design Results

The efficiency of our model was an initial concern, especially given our reliance on a physics-based method to estimate interaction energies. However, the runtime is ultimately quite competitive when compared against EvoPlay's AF2-based model. On standard Google Colab GPUs, running HERN and the OpenMM refinement takes about the same time as AF2. On the other hand, FoldX runs considerably faster than the time it takes to calculate the reward function from AF2 features. Some of this efficiency may be due to the relaxation performed by OpenMM. A single MCTS simulation runs about 10 seconds faster on our model compared to the EvoPlay AF2 model. With 10 simulations per action and 5 episodes, a run of our model takes, on average, 20 to 30 minutes, depending on the length of each episode.

In order to analyze the performance of our model, we compared the interaction energies of structures generated from our model to structures generated by HERN on its own. From the 10 structures that we tested, we found that our model generally performed better than our HERN baselines.

Out of the 10 antibodies we tested our model on, our model produced better, or more negative, interaction energies than HERN for nine of the structures (Table 1). Our model produced structures with interaction energies 4.28 kcal/mol lower on average. This difference slightly increases when only considering structures with relatively high native energies, increasing to 4.91 kcal/mol for structures with positive native energies. We also see that our model always converges to a structure with a more favorable, negative interaction energy. On the other hand, HERN failed to do so for eight of the samples, even on structures with low native energies. These results suggest that our model greatly outperforms our baselines, demonstrating the significant improvement that our model provides in sequence generation. Figure 4 depicts generated paratopes for 1NCB, a neuraminidase-antibody complex.

PDB	Native	HERN	EvoPlay
4ffv	TRFRDVFFDV	ARDYYGYFDV	TCDRDVFFSV
	0.333828	0.683736	-2.63029
3cx5	ARSEYYSVTGYAMDY	ARGYYYGYGYGYFDV	DRSEYKFVTGYAQDF
	-3.75909	0.940481	-6.20123
5mes	ARQVGATWAFDI	ARGGYYYGYFDV	DRHVGYTQAWDI
	2.04656	0.758949	-4.37634
1ncb	ARGEDNFGSLSDY	ARDYYGYGYGFDV	ARQEMNFGILIDY
	-7.73192	1.0437	-3.08202
1n8z	SRWGGDGFYAMDY	ARGYYGYGYWFDV	SRNGGDGPYAFDF
	-4.38252	-9.29175	-6.13511
4g6j	ARDLRTGPFDY	AREGDYGYFDV	AFDPITGPFDY
	0.457618	0.0637	-3.68364
3uzq	SRGWEGFAY	ARDGYGFDY	QRCPWGHGG
	2.15188	0.305946	-3.14314
3w9e	ARDLTTLTSYNWWDL	ARLYYYGVYWYGMDV	ARDKTTLFSYNWWDY
	-5.14338	-2.25919	-7.74877
3k2u	AKWWAWPAFDY	ARDYYYGYFDV	ANEVAWPAFGY
	-4.49475	0.307115	-4.37354
2dd8	ARDTVMGGMDV	ARDRGYGYFDV	ARNTVPGGANV
	1.26538	4.99474	-3.89286

Table 1: Generated sequences and computed energies. Energies in units of kcal/mol.

Figure 4: Comparison between generated structures. PDB 1NCB. Native paratope in orange (*E* = -7.73). HERN paratope in yellow (*E* = 1.04). EvoPlay paratope in magenta (*E* = -3.08).

It is interesting to note that our results were a significant improvement over HERN, given the number of mutations made per structure. Our model made relatively few mutations for each structure, usually not exceeding 5, while sequences generated by HERN differed greatly from the native sequences. Moreover, the small number of mutations to each structure in our model led to the generation of better structures than HERN. Our results demonstrate a potential advantage of models based on directed evolution: since native sequences are already optimized by natural selection, these serve as a good starting point when searching for improvements.

There are a few limitations that make the current reinforcement learning model difficult to use in practice. It is important to note that our reward function is based on an approximation of the true binding energy. Generated sequences are also highly variable, usually differing greatly between runs. This variance can likely be attributed to the limited exploration performed by the model. With the parameters that we used, around 50 unique sequences are sampled in each run. Increasing the number of simulations and episodes would increase the search space. Randomness in the OpenMM relaxation also leads to considerable variation in computed interaction energies; therefore, tuning various OpenMM parameters could produce more consistent structures.

Future Goals

Moving forward, there are many things that we would like to see built upon our work described within this report. The first of these deals with optimizing the runtime of the model, allowing for more rounds of reinforcement learning. Currently, the main limiting factor with respect to the model's runtime is the OpenMM relaxation model that HERN applies for correcting its structure predictions. This part of the model takes nearly nine seconds to run per structure, significantly limiting the amount of structures we are able to work with. If allotted more time, we hope to work on optimizing the speed of the docking and relaxation model, improving this limiting runtime while also maintaining the accuracy of the corrections.

Once the limitations with runtime are alleviated, we also hope to expand the datasets used in testing and training. Currently, only around 50 structures are used for training, significantly limiting the accuracy of our model. By expanding the dataset further and training on a significantly larger training set, we will be able to make the model more accurate and be able to trust our results more.

We also look to analyze if we can use DSMBind (Jin et al., 2023) as a reward model. DSMBind uses a different metric than HERN to predict structures, which has proven to be a much more accurate indicator of antibody-antigen interactions in comparison to computed interaction energies. The main barrier against us currently using DSMBind is that it takes in structures instead of sequences as input, and it would have been too time consuming for us to have generated structures for our model given our limited time frame. However, given more time, we believe we could further improve the performance of our model by incorporating DSMBind into the reward model.

Finally, it is important to experimentally validate our generated sequences. Predicted sequences are docked in isolation, and only the CDR-H3 is modeled. Experimental binding assays are needed to confirm if our computed energies reflect the true structures. Furthermore, it is unknown whether these sequences and antibodies can be synthesized and put to practical use.

Comparison with Original Proposal

The model proposed in this paper only differs from our original proposal in the reward model used. Our original proposal used DockGPT (a transformer-based model that is able to perform protein docking in a flexible and site-specific manner); however, due to a lack of code availability, we replaced DockGPT with HERN. This issue could have been foreseen earlier had we tried to verify the usability of our dependent models. During our original project proposal review, the main concern we had was the project's feasibility, given the high runtimes of our suggested models. Our suggested models ended up having long runtimes, ranging from 20-30 minutes on average per structure. Instead of the energy computation, as we had expected, the main bottleneck came from refining the docked structures. To compensate, we limited the number of structures we used for testing. Due to this update, we were able to complete all of our original goals, producing an updated EvoPlay that uses HERN as a reward model and incorporating interaction energies into our predictions. Having finished our project, we believe our original proposal was not overly ambitious, as we could complete the project without making any major updates to our specific goals.

Commentary on Experience

Our experience on the project generally went well. All of our structures came from a dataset provided with HERN, which meant that retrieving and utilizing the dataset was quite simple. The only issue with the relatively small testing dataset. Part of this comes from the limited number of antibody structures in the Protein Data Bank. Since our model was slow and we had limited time, this turned out to be acceptable; however, if allotted more time, we could have improved our model by having a larger dataset of structures.

Our team also had a good experience managing our research and writing. A lot of the writing had already been completed from the project proposal, such as the background and implementation explanations, which meant that we could dedicate more of our time to working on our project. We were still able to make consistent progress on our report once we started the project by efficiently managing our writing time. There were many times when a team member was dependent on another member's work, so those blocked members would contribute to writing while waiting to be unblocked. This allowed us to get significant portions of the report completed before many parts of the project were finished.

The project itself had a few difficulties. The most challenging aspect of this project was setting up EvoPlay to integrate HERN as a reward model. In order to run EvoPlay, a significant amount of external packages had to be downloaded. Furthermore, EvoPlay was developed on some outdated packages, so we had to edit some of the code to be compatible with the latest package version. We also encountered difficulties running the models due to our lack of compute power. Both of the models relied on methods that were specific to NVIDIA GPUs. However, we were eventually able to get both models running on Google Colab. Loading the repositories and installing packages was tedious, but there were no issues using the Colab GPUs. Despite all of the difficulties, getting our EvoPlay model set up was also the most rewarding aspect of the project. After spending a significant amount of time fixing issues with running the model, we felt great finally getting the model to produce structure predictions.

Division of Labor

Ananth worked on adapting EvoPlay’s framework and overseeing the testing process. Wilson helped with the adaptation of EvoPlay and worked on adapting HERN to support iterative mutations. Mohamed helped retrieve baseline structures from our adapted HERN model and analyzed the results of the model in comparison to the baseline. Overall, our collaboration experience went well, particularly with our

communication. We used group chats and consistent meetings to work through implementation and set up questions, which greatly supported our completion of the project. The worst aspect of collaboration was probably setting up meetings, as our team members had different and busy schedules. Despite this, we were still able to communicate consistently. If we were to start over again, our advice would be to start a bit earlier. Although our project was successful, and we had enough time to complete it, there were certain times when it would have been difficult to recover had we encountered any major issues with our implementation.

Code

Modified EvoPlay repository and saved test structures:
drive.google.com/drive/folders/11DQVOOkzbGB575H3L1BWDUEWl4LlBVC2?usp=sharing

References

Delgado, J., Radusky, L. G., Cianferoni, D., and Serrano, L. 2019. Foldx 5.0: working with RNA, small molecules and a new graphical interface. Bioinformatics, 35(20):4168–4169.

Dunbar, J., Krawczyk, K., Leem, J., Baker, T., Fuchs, A., Georges, G., Shi, J., and Deane, C. M. 2014. SAbDab: the structural antibody database. Nucleic acids research, 42(D1):D1140–D1146.

Esquivel-Rodríguez, J., Yang, Y. D., and Kihara, D. 2012. Multi-LZerD: Multiple protein docking for asymmetric complexes. Proteins 80 (7), 1818–1833. doi:10.1002/prot.24079

Evans, R., O'Neill, M., Pritzel, A., et al. 2022. Protein complex prediction with AlphaFold-Multimer. bioRxiv 2021.10.04.463034; doi:10.1101/2021.10.04.463034

Jin, W., Barzilay, R., Jaakkola, T. 2022. Antibody-Antigen Docking and Design via Hierarchical Equivariant Refinement. arXiv 2207.06616; https://arxiv.org/pdf/2207.06616

Jin, W. Chen, X., Vetticaden, A., Sarzikova, S., Raychowdhury, R., Uhler, C., Hacohen, Nir. 2023. DSMBind: SE(3) denoising score matching for unsupervised binding energy prediction and nanobody design. bioRxiv 2023.12.10.570461; doi:10.1101/2023.12.10.570461

McPartlon, M., Xu, J. 2023. Deep Learning for Flexible and Site-Specific Protein Docking and Design. bioRxiv 2023.04.01.535079; doi:10.1101/2023.04.01.535079

Wang, Y., Tang, H., Huang, L. et al. 2023. Self-play reinforcement learning guides protein engineering. Nat Mach Intell 5, 845–860. doi:10.1038/s42256-023-00691-9

Appendix

PDB	Native	HERN
4ffv	TRFRDVFFDV	ARDYYGYFDV
3hi6	ASSYDFWSNAFDI	ARLYYGYYWYFDV
4cmh	ARGDYYGSNSLDY	ARDYYDYGYGFDV
5hi4	ARDLIHGVTRN	AREYGYGYFDV
3cx5	ARSEYYSVTGYAMDY	ARGYYYGYGYGYFDV
3nid	VRPLYDYYAMDY	ARGYYYGYAMDY
5mes	ARQVGATWAFDI	ARGGYYYGYFDV
1ic7	ANWAGDY	ARDYFDY
1ncb	ARGEDNFGSLSDY	ARDYYGYGYGFDV
5bv7	AREGAAVRSFYYSYYGMDV	ARLYYYSYYYGYGYYDMDV
4lvn	VIYRYDGQWVFDD	ARDYYYYYGYFDV
4ki5	TRTSYYFGSSYDFDV	ARGYYYGYGNWYFDV
5ggs	ARRDYRFDMGFDY	ARSYGYGYWYFDV
4fqj	ARDVQYSGSYLGAYYFDY	ARARLGYGYYGYWYAMDY
4etq	TRSNYRYDYFDV	ARLYYYGYAMDY
2adf	ARDNPYYALDY	ARERDYGYFDV
5nuz	TRRNTLTGDYFDY	ARYGYGYGYAMDY
4ydk	VRVTFYHEGSGYYYRAGNYFDS	ARDRRVYGYGYWYDYYYGGFDY
3rkd	ARIKSVITTGDYALDY	ARQGKYYGSYYGYFDV
2cmr	ARDNPTLLGSDY	ARDYYGYGYFDV
5b8c	ARRDYRFDMGFDY	ARDYGYGYGYFDY
3bn9	ARPYLTYPQRRGPQNVSPFDN	AREKLYARLSYYYYDYYGMDV
4xnq	ARVALFDILTGGWFDP	ARLYYYYGYTYYAMDV
2xqy	GRLGYVYGFDY	ARDGYYGYFDV
1a2y	ARERDYRLDY	AREGYGYFDV
3o2d	AREKDNYATGAWFAY	ARDFGYGYDYYAMDV
1fe8	AGNYYGMDY	ARDYGYFDV
1n8z	SRWGGDGFYAMDY	ARGYYGYGYWFDV
4g6j	ARDLRTGPFDY	AREGDYGYFDV
2ghw	ARDRSYYLDY	ARDYYGYFDV
2b2x	TRGFGDGGYFDV	AREGDYGYAMDY
1osp	ARSRDYYGSSGFAF	ARLYYGYDYWYFDV

PDB	Native	HERN
1uj3	ARDSGYAMDY	ARDYYGYFDV
3ffd	ARQTTMTYFAY	ARDYGYGYFDY
3uzq	SRGWEGFAY	ARDGYGFDY
4h8w	AKDLRLGGGSDY	ARDYDYGYGFDY
2vxt	ARGLRF	ARDFDY
4dvr	AADPWELNAFNV	ARDYYDYGYFDV
4qci	ARHPYWYGGQLDL	ARDYYYDYGYFDV
3s35	TRHDGTNFDY	ARDGYGYFDV
3w9e	ARDLTTLTSYNWWDL	ARLYYYGVYWYGMDV
5f9o	VRNVGTAGSLLHYDH	ARGRYYGYYYGYFDV
1iqd	AVPDPDAFDI	ARDGYGYFDV
5d96	ASDSMDPGSFAY	ARGYYGYGYFDV
4g6m	ARNRYDPPWFVD	ARDYYDYGYFDV
4ot1	ARDGAKTVSNSGLSLLYYHNRLDA	ARARGYYYGDYPRSYDYDYYGMDV
5j13	ARAPQWELVHEAFDI	ARDLYGYGYWYAMDV
5l6y	ARDSSSSWARWFFDL	ARLYYYGDVSYAMDV
3l95	ARGSGFRWVMDY	ARDYGYGYGMDV
2xwt	VGLDWNYNPLRY	ARDGYGYGYFDV
3k2u	AKWWAWPAFDY	ARDYYYGYFDV
4dtg	ARLGGYDEGDAMDS	ARGYYYDGYYYFDV
5en2	ARRRVYYGSNYIYALDY	AREGSYYGDYFYDYFDV
5d93	ARSEDWFAY	ARDYGYFDY
3mxw	ARDWERGDFFDY	ARLYYGYGYFDV
2dd8	ARDTVMGGMDV	ARDRGYGYFDV
2ypv	VRDYYGSSYFDY	ARERGDYYAMDY
1a14	ARSGGSYRYDGGFDY	ARQDYYDYYYGYFDV

Figure A1: HERN-generated sequences.