Better Prior Distribution for Antibody Complementarity-Determining Regions Design

Jun Wu[1], Xiangzhe Kong[2,3], Ningguan Sun[4], Jing Wei[5], Sisi Shan[5], Fuli Feng[1], Feng Wu[1], Jian Peng[4], Linqi Zhang[5], Yang Liu[2,3*], and Jianzhu Ma[3,6*]

[1]University of Science and Technology of China, Hefei, China
[2]Department of Computer Science and Technology, Tsinghua University, Beijing, China
[3]Institute for AI Industry Research, Tsinghua University, Beijing, China
[4]Helixon US Inc, USA
[5]Comprehensive AIDS Research Center, Center for Infection Biology, Pandemic Research Alliance, School of Basic Medical Sciences, Tsinghua Medicine, Tsinghua University, Beijing, China
[6]Department of Electronic Engineering, Tsinghua University, Beijing, China
[*]Correspondence should be addressed to:liuyang2011@tsinghua.edu.cn, majianzhu@tsinghua.edu.cn

ABSTRACT

Designing antibodies with desired binding specificity and affinity is vital for various fields, including pharmaceutical and biological research. While recent advances leverage diffusion-based models for co-designing the sequences and structures of the Complementarity-Determining Regions (CDRs), they are still confronted with three challenges: non-informative prior distribution, incompatibility with discrete amino acid types, and impractical computational cost in large-scale sampling. To address these pitfalls, we proposed FlowDesign, a novel approach for sequence-structure co-design based on Flow Matching with the following merits: (1) Flexible selection of prior distributions; (2) Direct matching of discrete distributions; (3) Enhanced computational efficiency for large-scale sampling. By leveraging various prior distributions, we first identified data-driven structural models as the most informative prior. Subsequently, evaluations against strong baselines illustrated the superiority of our model across diverse metrics including Amino Acid Recovery (AAR), RMSD, and Rosetta energy. We also demonstrated the utility of FlowDesign in designing antibodies to target HIV-1 cellular receptor CD4. Through Biolayer interferometry (BLI) and pseudovirus neutralization evaluation, we obtained regenerated antibodies with better binding affinity and neutralizing potency compared with the state-of-the-art HIV antibody Ibalizumab across multiple HIV mutants at the cellular level. These results highlight the power of our FlowDesign model in antibody design and its potential application to engineering other protein molecules.

1 Introduction

Antibodies are essential to the immune system for their ability to neutralize pathogens[1] by binding to specific antigens. Designing antibodies that can target particular antigens has significant implications for pharmaceutical and biological research[2]. The Complementarity-Determining Regions (CDRs) are pivotal in the functional repertoire of antibodies, as they constitute the majority of the binding area. Therefore, the primary focus of antibody design is to identify the CDRs with favorable binding affinity and specificity to the antigen. However, it is not feasible to traverse the entire combinatorial space of CDRs via laboratory experiments due to the unaffordable cost, thus efficient exploration with computational methodologies becomes crucial.

Conventional methods sample sequences and structures through simulations with empirical energy functions[3–6], which are time-consuming and prone to be stuck in local optima. Recent progress

exploits deep-learning models to design CDRs or the entire antibody structure[7–12]. Protein language models (PLM), akin to their counterparts in natural language processing, offer a transformative methodology for designing antibody sequences[13,14]. PLM learns the grammar of protein sequences by training on vast datasets of known proteins, thereby capturing the intricate patterns among amino acids that determine protein structure and function. When applied to the task of antibody design, PLM can predict sequences for the CDRs that are likely to bind with high specificity and affinity to target antigens[15–17]. The use of PLM in antibody design significantly accelerates the discovery process, reducing the reliance on exhaustive wet lab experimentation. By efficiently narrowing down the vast search space to a manageable subset of promising candidates, these models facilitate a more focused and cost-effective empirical validation process.

While they showed potential when adapted to the antibody domain[15–17], the lack of structural modeling made them suboptimal[7,9]. A critical issue of sequence-based methods is that they can only enhance the stability of antibodies based on the sequence of individual antibodies. They struggle in improving the affinity between antibodies and antigens because they are unable to describe the sequence or structure of antigens. Another category of promising methods is diffusion models for sequence-structure co-design[11], which not only model the intricate relations between sequences and structures but also enable diverse sampling of antibodies under specified conditions. Diffusion-based models first inject noises into the data in the forward process, resulting in a prior distribution which are usually standard Gaussian, then learn a generative reverse process to reconstruct the data distribution by denoising the noisy states. In the context of antibody design, the application of diffusion models on both sequences and structures of the CDRs excels in capturing the joint distribution and generates antibodies with enhanced binding affinity[11,12,18].

Despite the encouraging progress, current diffusion-based antibody-design models still suffer from multiple drawbacks. First, the prior distribution, which is commonly a standard Gaussian, fails to reflect the physical plausibility and feasibility of antibody molecules. CDRs consist of continuous linear loop structures constrained by structurally conserved frameworks, with certain position-specific amino-acid preferences. While other deep-learning-based[8,15,17] or energy-based models[3–6] can easily integrate these prior knowledge into their model initialization, current diffusion models overlook these aspects with the prior distribution limited to standard Gaussian distribution. Second, the continuous nature of the diffusion algorithm is barely compatible with the discrete nature of amino acid types. Adding noises leads to smooth transitive processes but applying it to amino acid types easily induces abrupt changes between consecutive steps, which hinders the model from capturing the desired sequence distribution precisely. Third, the iterative denoising procedure incurs impractical computational costs for large-scale sampling. Diffusion models generate each sample through T denoising steps, necessitating running the model for multiple iterations. Therefore, generating large-scale samples for screening potential candidates becomes computationally challenging due to the increased computational cost.

To address the issues, we developed a new algorithm, named FlowDesign, to tackle antibody sequence-structure co-design tasks based on Flow Matching[19]. Unlike traditional diffusion-based models, our model approached CDR design as a Transport Mapping problem, which learned a direct mapping from arbitrary initial distributions to the target distribution. Resembling a one-step "diffusion" model, it naturally enjoys the following desirable merits: (1) Flexibility in the selection of the prior distribution, which allows integration of diverse prior knowledge; (2) Direct matching of discrete distributions, which avoids non-smooth generative processes on the amino acid types; (3) High efficiency, which facilitates large-scale sampling with low computational cost. We first assessed the impact of prior knowledge in different forms, including distributions derived from

protein language models, physicochemical functions, and structural data-driven models[7], among which the structural data-driven models yielded the best prior distribution. Next, we conducted a comprehensive evaluation of the sequences and structures of antibodies designed by different computational methodologies. Results demonstrated that adding proper prior knowledge could almost always improve performance with respect to different metrics. In addition, we evaluated the performance of FlowDesign in designing antibodies at both the molecular and cellular levels against Human Immunodeficiency Virus Type 1 (HIV-1). HIV-1 has caused a significant threat to human health. Broadly neutralizing antibodies (bnAbs) against HIV-1 are promising candidates for HIV-1 prevention and treatment. However, high-frequency mutations and a large number of resistant strains of HIV-1 make it hard to discover antibodies through traditional antibody screening methods. Therefore, there's an urgent need to use the AI model to assist in the design of antibodies against HIV-1. We used the FlowDesign model to redesign the HCDR3 of the HIV-1 antibody Ibalizumab (imab), which targets the HIV-1 cellular receptor CD4. We evaluated the designed antibodies through SPR and HIV-1 pseudovirus neutralization assay and found multiple candidates have comparable or even better binding affinity and neutralizing compared with the state-of-the-art HIV antibody Ibalizumab[20]. These results showed that our model has the ability to regenerate antibodies and has potential applications in engineering other protein molecules.

2 Results

Overview of FlowDesign

We represented an antibody as $\{\mathbf{s}_i, \mathbf{x}_i, \mathbf{O}_i\}$, namely the amino acid type, the C_α atom coordinate and the backbone orientation of each residue in the antibody(Figure 1.A). The inference process begins with a given antibody-antigen complex with CDRs to be designed. The CDRs are initialized from a certain prior distribution. There are different options for prior distributions, which we discuss in the next section. Subsequently, the model predicts drift forces on each residue from the inputs, which are interpreted as the transformation from the initial distribution to the target distribution. Given an informative initial distribution, we can directly perform a one-step transformation to achieve the target distribution. However, if delicate refinements are required, we can also simulate an integration process by scaling the predicted drift force with a coefficient and repeating the cycle for N times, where the selection of N depends on the application scenarios. After the inference, we adopted the side-chain packing function in the Rosetta package[5] to construct the side chains and perform relaxation to prevent inter-atomic collisions. As the side-chain packing software we adopted only produces a small portion of atom clashes. During the sampling stage, we directly discarded structures with atomic clashes. As for training, we exerted supervision on the drift forces to update the parameters of the model. Specifically, for each antibody-antigen complex in the dataset, we calculated its difference from an initial complex sampled from the prior distribution as the ground truth for drift forces. Then we minimized the distance between the predicted drift force and the ground truth one given the same initial complex. Unlike other diffusion-based models, which repeatedly add and remove noise, our training objective is a one-step transformation, directly matching the initial distribution with the target distribution. FlowDesign offers two design approaches: The first mode designs the antibody CDR given the framework and antigen complex. The second mode fixes the framework's amino acid type and structure, employs dyMEAN[8] and AlphaFold3[21] for complex prediction, and subsequently designs the CDRs.

Performance for different prior distributions

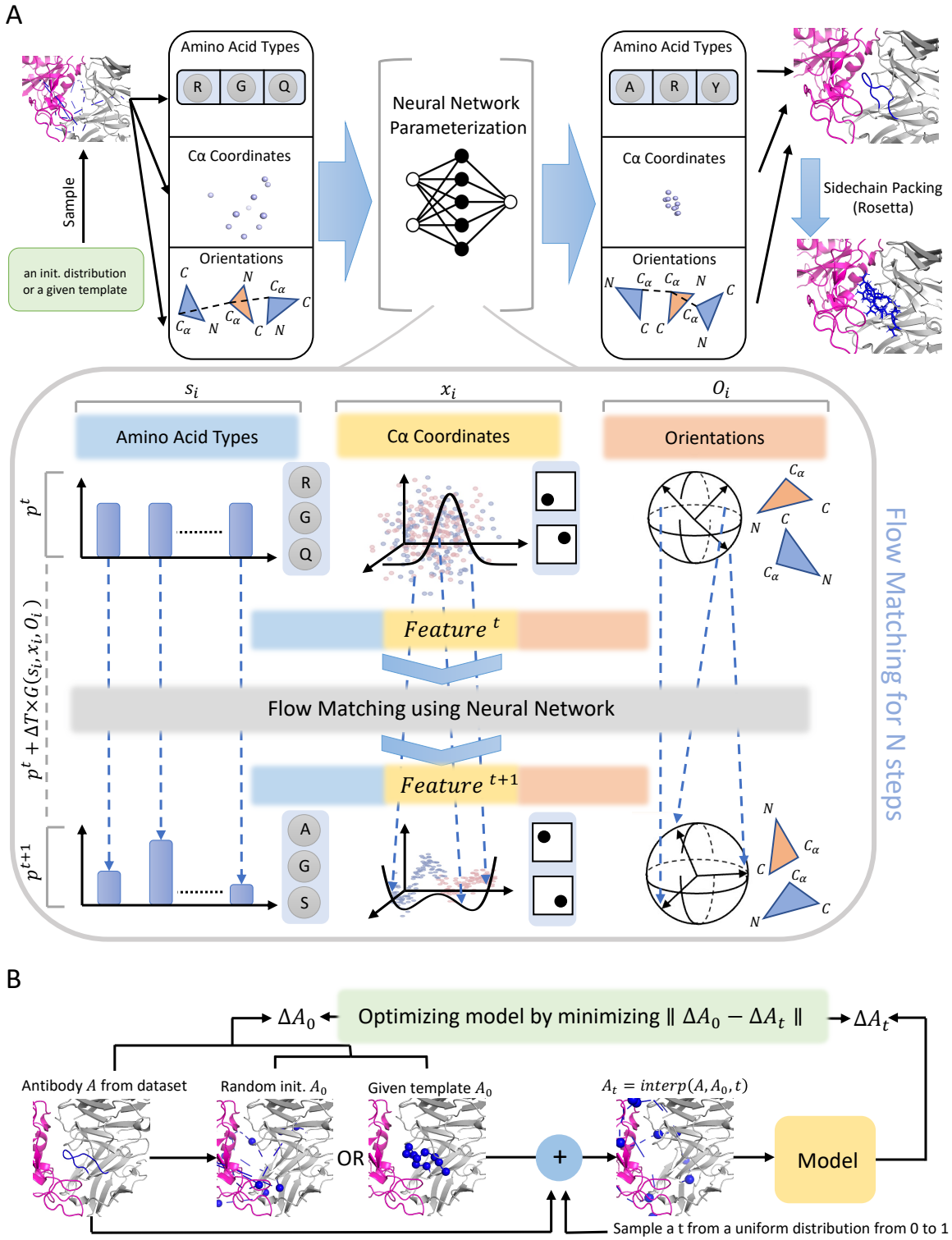


Figure 1. Inference and training process of the FlowDesign model. **(A)** FlowDesign learns to generate antibodies by matching the prior distribution to the data distribution extracted from the antibody dataset. **(B)** In the training process, the model parameters were trained by minimizing the difference between the initialized distribution and the target distribution.

In contrast to conventional diffusion-based models, our model enjoys the flexibility to integrate various types of prior distribution into the generation process. To assess the impacts of different prior distributions and find the optimal strategy, we conducted attempts on four distinct prior distributions: random distribution, the sequence-structure joint distribution outputted by an all-atom model dyMEAN[8], the structure distribution generated by the kinematic-closure-based method KIC[22,23], and the sequence distribution outputted by a large protein language model ESM2[13]. The model was trained and validated on CDRH3 with the four prior distributions because of the critical role in antigen-antibody binding and the high diversity of CDRH3. Experimental results revealed that using dyMEAN as the prior distribution generated antibodies with the best quality. In particular, the model utilizing dyMEAN achieved the highest sequence recovery rate, showing a nearly 15% improvement over methods using random initialization (Figure 2A). Protein language model ESM2 struggled with the high diversity of its generated antibody sequences, leading to a lower sequence recovery rate. It could not provide effective prior knowledge for the modeling of antibody structures due to the lack of structural information. The model with dyMEAN prior obtained the lowest average RMSD (root-mean-square deviation), at 2.291Å on average, consistently outperforming the other three methods across various sampling ratios (Figure 2B). This indicated that the predicted conformations were the closest to the original antibody structures. KIC initialization reached relatively poor performance among all the methods, due to the limited number of solutions provided by the kinematic equations due to the unaffordably high computational cost (STAR Methods). In addition, due to the effective sequence and structural knowledge provided by dyMEAN initialization, the generated CDRs had lower energy compared to those from other methods, resulting in a more stable overall binding (Figure 2C). We further provided the visual results of antibody-antigen complexes generated by the dyMEAN-initialized model (Figure 2D), which exhibited better stability compared with results from other methods.

Performance for Sequence-Structure Co-design

To validate the superiority of our model, we compared it against multiple strong baselines for antibody design, including HERN[10], Refinegnn[9], RosettaAb[5] which is based on empirical energies, the diffusion-based model Diffab[11], and dyMEAN[8], across diverse metrics. We removed all structural and sequence data for the antibody CDRs and simultaneously designed all CDRs in all H-chains and L-chains together. Notably, the RefineGNN and HERN algorithms do not accept light chains as input, and HERN can only generate sequences. Therefore, we did not test the metrics of these methods on light chains. The experimental results indicated that our model achieved a higher amino acid recovery rate on CDRs compared to other methods due to the effective incorporation of prior knowledge (Figures 3A-F). Notably, our model surpassed other methods by a large margin on CDRH3, which is also the most critical part of binding among all the CDRs. By selecting the top 1% of sampled antibodies based on energy, the amino acid recovery rate on CDRH3 could exceed 60%, which exhibited inspiring potential for enhanced efficacy and efficiency in practical applications. We noticed that our model's performance improvement is limited at different sampling ratios, with CDRH3 amino acid recovery rate increasing by less than 5% when expanding the sampling range. This is because we achieved sufficiently good conformations within the top 1% of the screening. Expanding the screening range did not significantly improve the metrics, demonstrating that our method can efficiently perform large-scale screening using energy criteria. Moreover, our model achieved a lower average RMSD on CDRs compared to most other methods except dyMEAN. The reason is that the original version of dyMEAN only generates one antibody while FlowDesign is a generative model which can generate multiple antibodies. Given the widespread adoption of

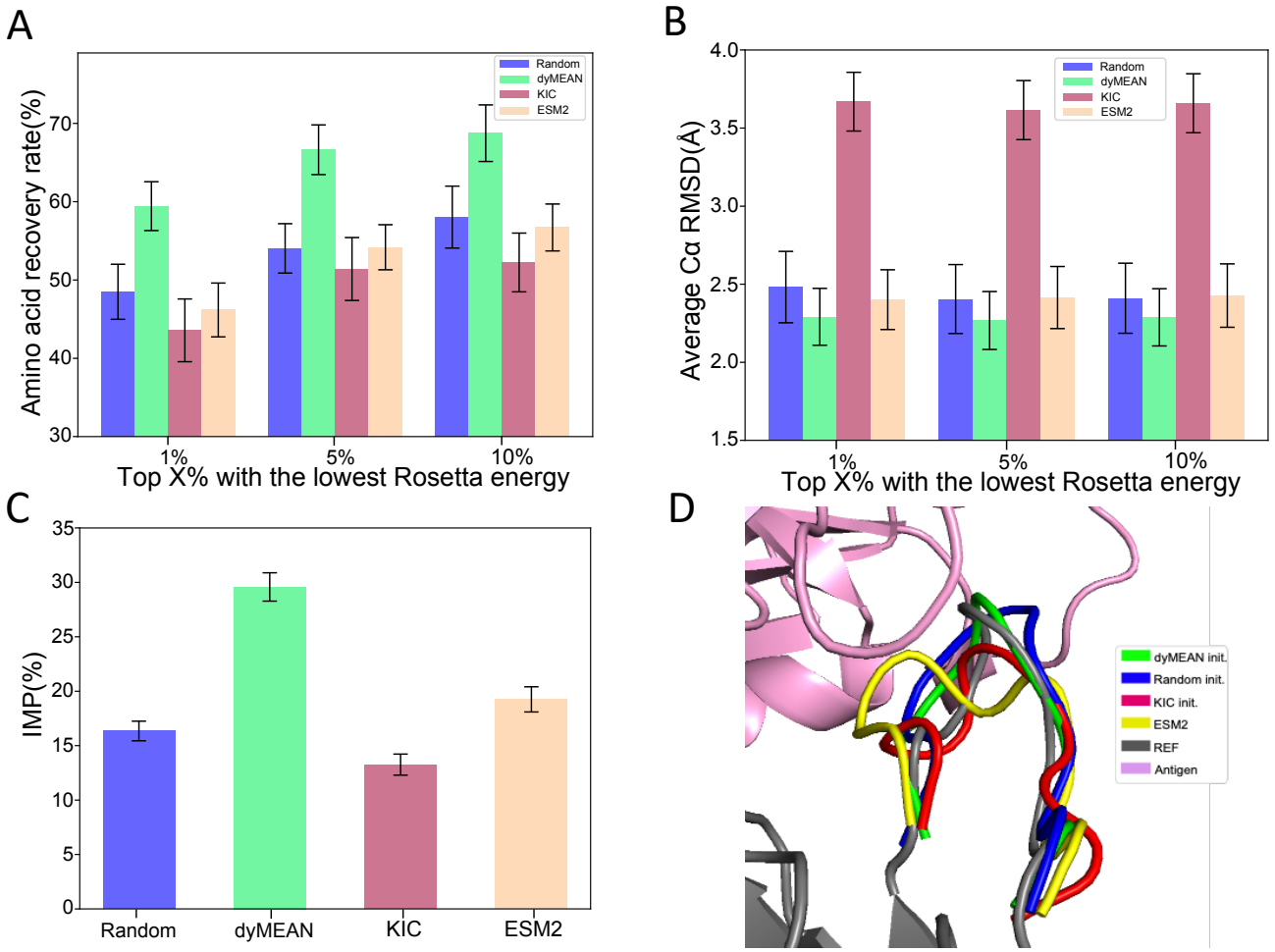


Figure 2. Performance for different prior distributions. **(A)-(C)** The performance of amino acid recovery rate, RMSD, and IMP for antibodies generated by different prior distributions, respectively. IMP represents the proportion of generated antibodies with energy lower than that of the original antibodies. Error bars represent 90% confidence intervals. **(D)** 3D structures of the antibody CDRH3 generated by different initialization methods, where the antigen structure is shown in pink and the original antibody structure is depicted in gray.

high-throughput display platforms, tools that enable the simultaneous design of multiple diverse antibodies are better suited to the practical demands of the biopharmaceutical industry. Specifically, it performed notably better on CDRH3, with an average reduction of 0.693Å compared to Diffab method (Figure 3G). We visualized an example of design by our model in Figure 3H, which illustrated that our model effectively transformed the initial distribution to the target distribution, resulting in an antibody-antigen complex that was more stable than the initial one.

Next, we conducted an extensive assessment of the antibodies generated by our model and Diffab, the only models possessing enough diversity in large-scale sampling among all the models. The antibodies sampled by our model exhibited a significantly lower energy landscape across all CDRs compared to those sampled by Diffab (Figures 4A,B). Additionally, the proportion of antibodies designed with lower energy than the native one is also higher, particularly with a 13.11% increase in CDRH1. This suggested our model was capable of providing more potential candidates than the diffusion-based model when the number of samples was limited. We showed the proportion of designed antibodies with better amino acid recovery than the prior distribution, namely, the distribution from dyMEAN, after energy ranking (Figure 4C). Apart from confirming the superiority of our model over Diffab, the results also confirmed that our model effectively integrated the prior knowledge provided and learned a more complete and accurate distribution based on prior knowledge. In addition, it could be observed that high-quality antibodies were more concentrated at the forefront of the energy ranking. Further, the point-wise comparison suggested that at different energy sampling ratios, our model consistently outperformed the Diffab in the restoration of both amino acids and structures (Figures 4D,E). The experimental results above demonstrated that our model surpassed other models in terms of designing stable and plausible CDRs, which was particularly notable for the typically challenging CDRH3. Furthermore, under the setting of large-scale sampling, our model was capable of yielding more candidates on the desirable energy landscape than Diffab within the same number of samples, exhibiting greater potential for application.

We explored the relationship between the RMSD of generated CDRs backbones and the energy of the final conformations. The trending in Figure S3A revealed that higher RMSD of the backbone corresponds to increased energy after side-chain packing. Two generated conformations with high and low backbone RMSD were visualized in Figures S3J,K. It demonstrated that lower backbone RMSD led to lower energy after side-chain packing and better side-chain alignment with the reference data. We also analyzed the influence of backbone RMSD on the side-chain atomic clashes. Results in Figure S3C indicated that the proportion of side-chain clashes during generation was barely related to the type of CDRs. Figures S3D-I visualized the relationship between backbone RMSD and side-chain clashes on different CDRs (i.e. H1, H2. H3, L1, L2, and L3), which did not show significant correlations.

Performance for designing CDRs based solely on sequences

In real-world scenarios of antibody design and optimization, the specific structure of an antibody might be unknown, and only the sequence is available. To further assess the potential of our method in designing antibody CDRs, we extended our method to design CDRs based solely on the sequence information of the antibody framework. The specific approach involves generating the structure of the antibody framework, excluding the CDRs, from the known sequence. Then, we used FlowDesign for designing the CDRs. We tested two methods to generate the structure: dyMEAN[8] and AlphaFold3.[21] We then compared our method, which utilizes known framework structures, with the sequences-based design method across various metrics. Figures 5A, B show

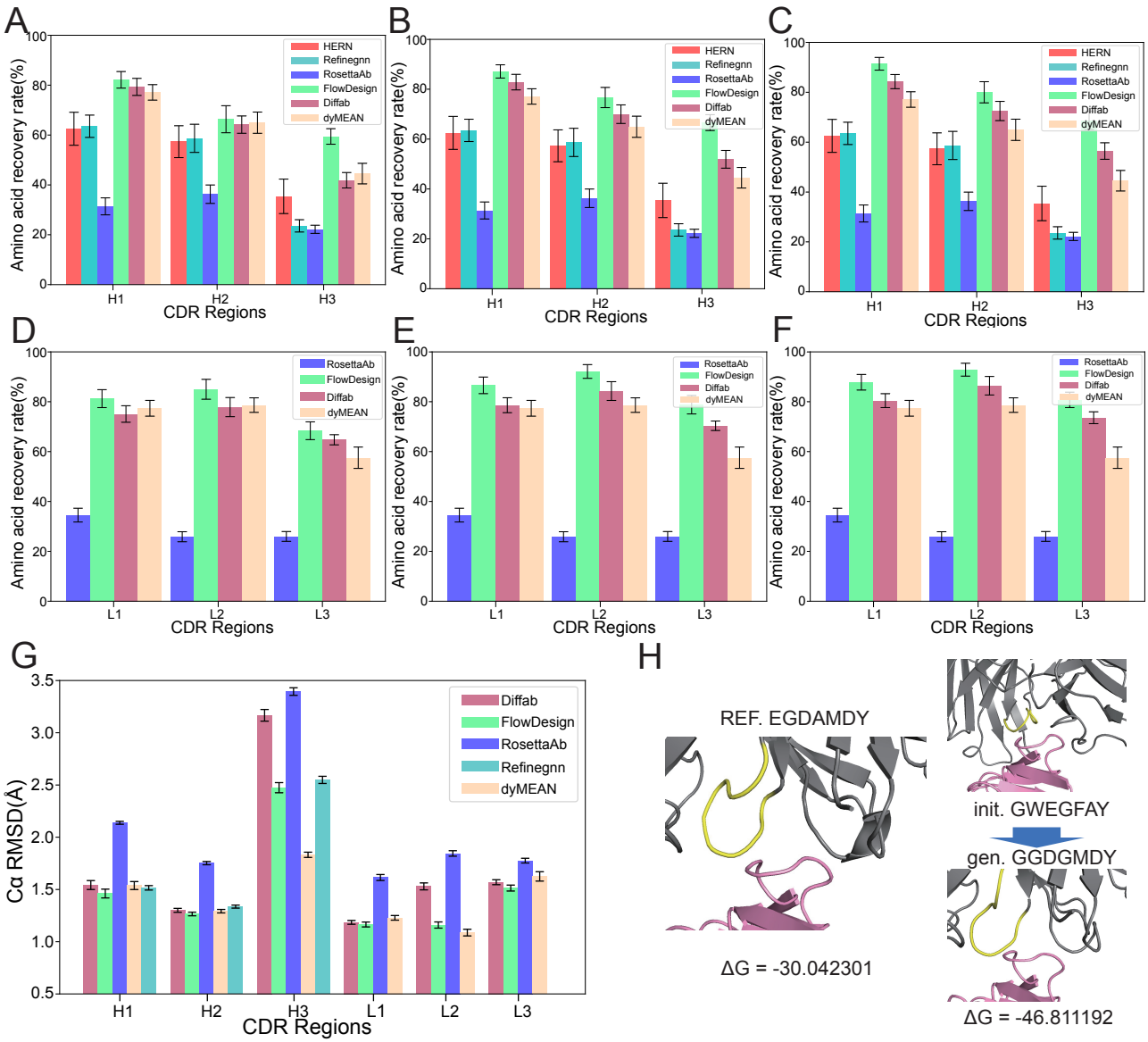


Figure 3. **(A)-(C)** Performance of different models on the amino acid recovery rate in different CDRs in heavy chains. From left to right, antibodies sampled are sorted by the Rosetta energy, and the maximum amino acid recovery rate is calculated for the top 1%, 5%, and 10% of these sampled antibodies. **(D)-(F)** Performance for different CDRs in light chains. From left to right, antibodies sampled are sorted by the Rosetta energy, and the maximum amino acid recovery rate is calculated for the top 1%, 5%, and 10% of these sampled antibodies. **(G)** Average C_α atom RMSD of the antibody structures generated by different models across various CDRs. Error bars represent 90% confidence intervals. **(H)** An example of a prediction made by our model. The left is the original antibody structure, and the right is the antibody predicted by the model from the initial distribution. The gray area represents the antibody framework, the yellow area is the CDRH3 to be generated, and the pink area is the antigen. ΔG refers to the binding energy between the antigen and the antibody.

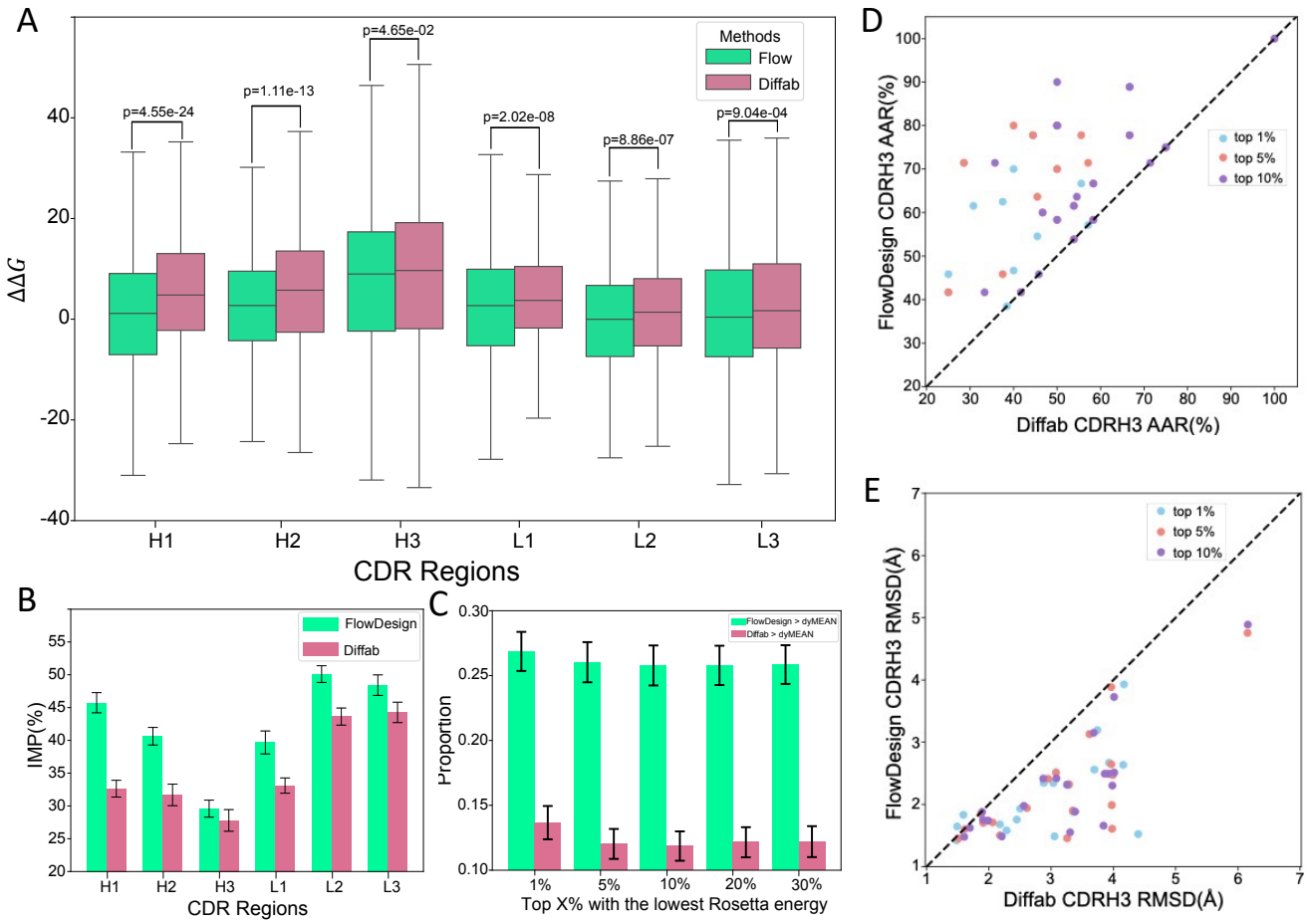


Figure 4. **(A)** The energy (The $\Delta\Delta G$) difference between the sampled conformation and the original native conformation. **(B)** The proportion of antibodies sampled by the two models that have lower energy compared to the original conformation. **(C)** The proportion of samples from FlowDesign and the diffusion-based model that have the amino acid recovery rate greater than that of the dyMEAN model. **(D),(E)** Performance comparison between FlowDesign and Diffab ranked by Rosetta energy. Error bars represent 90% confidence intervals. P values were computed based on a one-sided paired t-test with n=1000.

the AAR and RMSD of CDRs sampled at different ratios based on energy. It is evident that, aside from cases where the true structure is known, using AlphaFold3 for framework structure prediction framework results in the most accurate sequences and structures for the generated CDRs. Similarly, Figures 5C, G indicate that the energy of structures generated by this method is relatively low, making them well-suited for large-scale sampling. Of most, the most optimal performance is achieved with the known real structure. Among sequences-based design methods, complexes generated using AlphaFold3 are more stable than those from dyMEAN, with performance very close to knowing real structures. Figure 5H also shows that the frameworks generated by AlphaFold3 have lower RMSD, further supporting this point. Figures 5I, J, K visualize the complexes generated by three methods: dyMEAN-generated framework structure, the real framework structure, and the framework structure generated by AlphaFold3. The figures show that dyMEAN's performance in reconstructing structures from sequences is weak, resulting in docking poses that significantly differ from the real poses, leading to higher energy. In contrast, the structures generated by AlphaFold3 closely resemble the actual structures, yielding results that are more aligned with calculations based on the real framework structures. The experimental results indicate that our method can effectively extend from just using the antibody framework sequence, with performance close to scenarios with known real structures. This demonstrates the robustness of our algorithm.

Performance for HIV antibody design

As HIV-1 antibody screening and design is a big challenge using traditional methods, we tried to use FlowDesign to design HIV-1 antibody. Given that CDRH3 primarily affects antibody binding, we fixed the framework of the HIV-1 antibody and redesigned the CDRH3 of the antibody. We used MMseqs2[24] to filter out antibodies with more than 50% sequence identity in CDRH3 to Ibalizumab (imab) from the dataset and retrained the model. First, we used the model to sample and generate CDRH3 variants under different noise intensities using the framework of antibody Ibalizumab (imab), producing around 50,000 different CDRH3 sequences. We then ranked the generated antibodies using Rosetta energy and selected 12,000 unique sequences for further experiments. A yeast display library was then constructed with these 12000 HCDR3 variants in the Ibalizumab backbone. Subsequently, fluorescence-activated cell sorting (FACS) was performed on the yeast display library based on antibody expression level and binding affinity to CD4. Increasing proportion of the yeast clones bound to CD4 was observed from 0.19% for the first sort (Figure 6A), to 25.9% for the second (Figure 6B) and to 61.7% for the third sorting (Figure 6C). Yeast displayed single clones Ibalizumab and FZD5 antibody 2919 were act as positive control and negative control (Figure 6D, 6E). After the yeast display, we performed next-generation sequencing on all screened antibodies, selected the top 10 antibodies with the highest reads, and chose the three antibodies with the lowest energy for further experiments, named imab-mut-1, imab-mut-2, and imab-mut-3.

The equilibrium dissociation constant (KD) with CD4 for these three antibodies were 0.19 nM, 0.56 nM, and 0.90 nM respectively. We assessed the broad-spectrum neutralization capacity by performing pseudoviral neutralization experiments against 11 HIV-1 pseudoviruses. These pseudoviruses are widely distributed, including subtype A, subtype B, subtype C, subtype CRF01, and subtype CRF07, and span from tier 1 to tier 3. Compared with Ibalizumab, the neutralization IC50 values of these three mutants for different pseudoviruses were comparable by orders of magnitude. All three antibodies had a better neutralization ability than the ancestral antibody to some of the pseudoviruses, but overall, imab-mut-2 performed best, followed by imab-mut-1, and imab-mut-3 performed relatively poorly. In the neutralization of 11 pseudoviruses, imab-mut-2 neutralized half of the pseudoviruses (TRO_11, X2278_C2_B6, BJOX002000_03_2, CNE4,

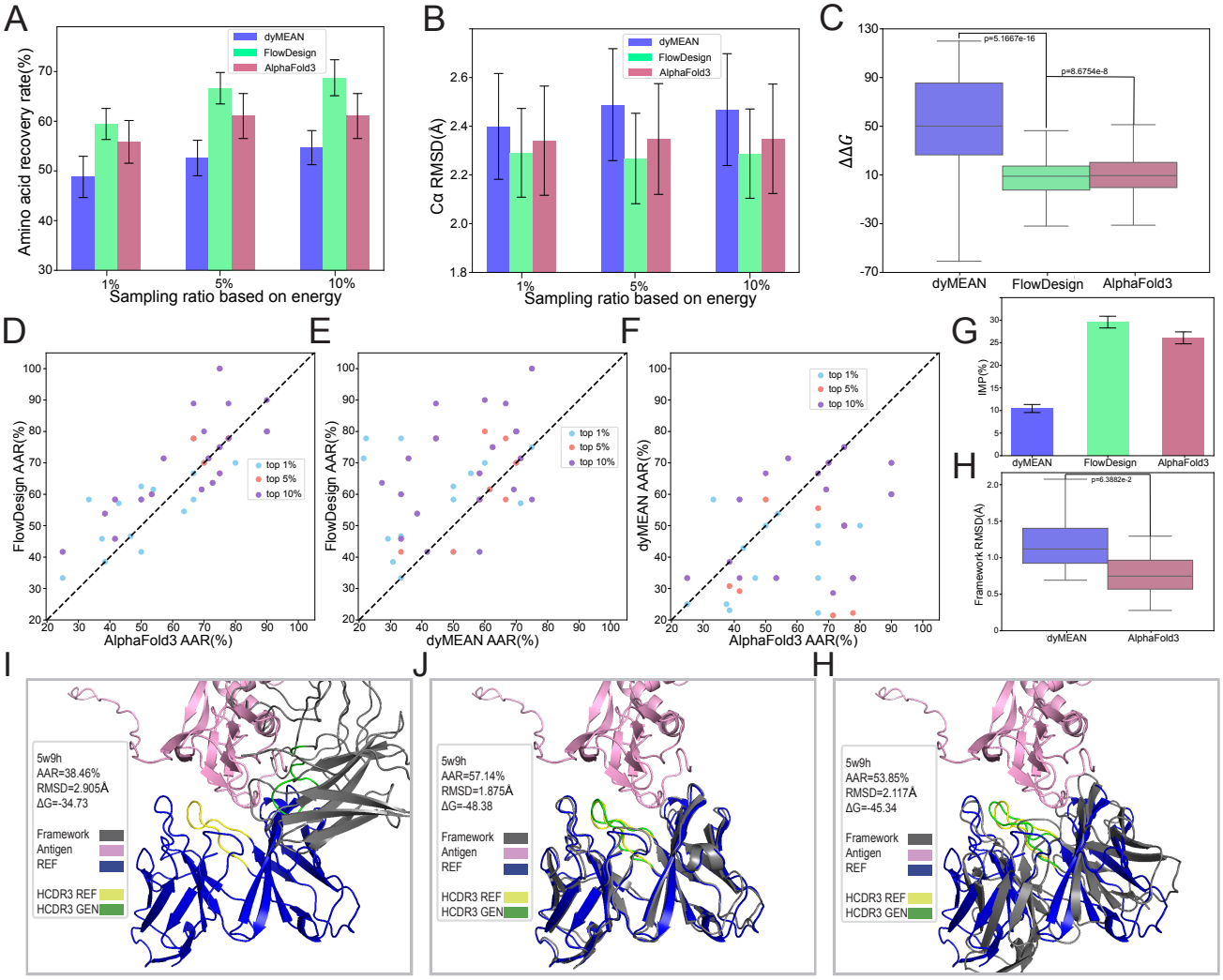


Figure 5. **(A)** Amino acid recovery rate of the dyMEAN generated framework, AlphaFold3 generated framework structure, and the known true structure for the FlowDesign method. **(B)** C_α RMSD of the dyMEAN generated framework, AlphaFold3 generated framework structure, and the known true structure for the FlowDesign method. **(C)** $\Delta\Delta G$ of the dyMEAN generated framework, AlphaFold3 generated framework structure, and the known true structure for the FlowDesign method. **(D)(E)(F)** Performance comparison between three different structure initialization methods ranked by Rosetta energy. **(G)** IMP of three structure initialization methods. **(H)** Comparision of RMSD between frameworks generated by dyMEAN and AlphaFold3. **(I)(J)(K)** Visualization of the generated complex using three structure initialization methods. Error bars represent 90% confidence intervals.

CNE14, SF162) better than imab-WT, as shown by the neutralization IC50 value of imab-mut-2 for CNE4 of 0.006 μg/ml, the IC50 value of the imab-WT is 0.024 μg/ml. Furthermore, the neutralization ability of imab-mut-2 to two of these pseudoviruses (398_F1_F6_20, CNE8) was comparable to that of Ibalizumab.

We visualized the energy of each residue in the CDRH3 from IMAB-WT and the model samples to identify patterns as shown in Figure S6. It can be observed that the last five amino acids of the CDRH3 (107A, 108W, 109F, 110A, 111Y) region in imab have maintained relatively low energy. Mutations in these five amino acids are likely to increase the conformation's energy. As shown in Figure S6, the three mutations (mut-1, mut-2, mut-3) with a similar Kd magnitude (1e-10) to imab-WT did not alter this region. However, in Figures S6 D, E, changes at positions 107 and 110 led to a significant energy increase in the residues of this region, resulting in an experimentally measured increase in Kd (1e-9). Based on the energy of imab-WT, the residues at positions 103, 104, and 105 show significantly higher energy than their surroundings. This suggests that modifying the residues in this region could potentially improve the binding affinity of imab-WT without affecting the stability of the surrounding regions. Figures S6A-C display three mutations generated by FlowDesign, imab mut-1, mut-2, and mut-3, all of which involve changes in this high-energy segment. It indicates that the energy levels of these residues decrease while the energy of the surrounding regions does not increase significantly. Experimentally, these mutations demonstrate affinities that are on par with or even better than imab-WT.

3 Discussion

In this work, we developed a model based on the Flow Matching algorithm which can map any prior distribution to the data distribution to design antibody CDRs. In our algorithm, it is important to note that the model relies on the complex structure of the target antibody and antigen. Incorrect structures may adversely affect the model's performance. However, recent breakthrough shows that the antibody-antigen complex structures can be accurately predicted given only their protein sequences[21]. Therefore, in the near future, it can be expected that an initial antibody can be obtained either by immunizing animals or using display technology, then the three-dimensional structure of the antibody-antigen complex is predicted by AlphaFold3, and finally the binding affinity could be enhanced through FlowDesign. In our future work, we will focus on how to obtain accurate structures of complexes in specific application scenarios while reducing the model's dependency on these complexes during inference. We believe this will enhance the applicability of the algorithm in practical antibody design tasks.

4 STAR METHODS

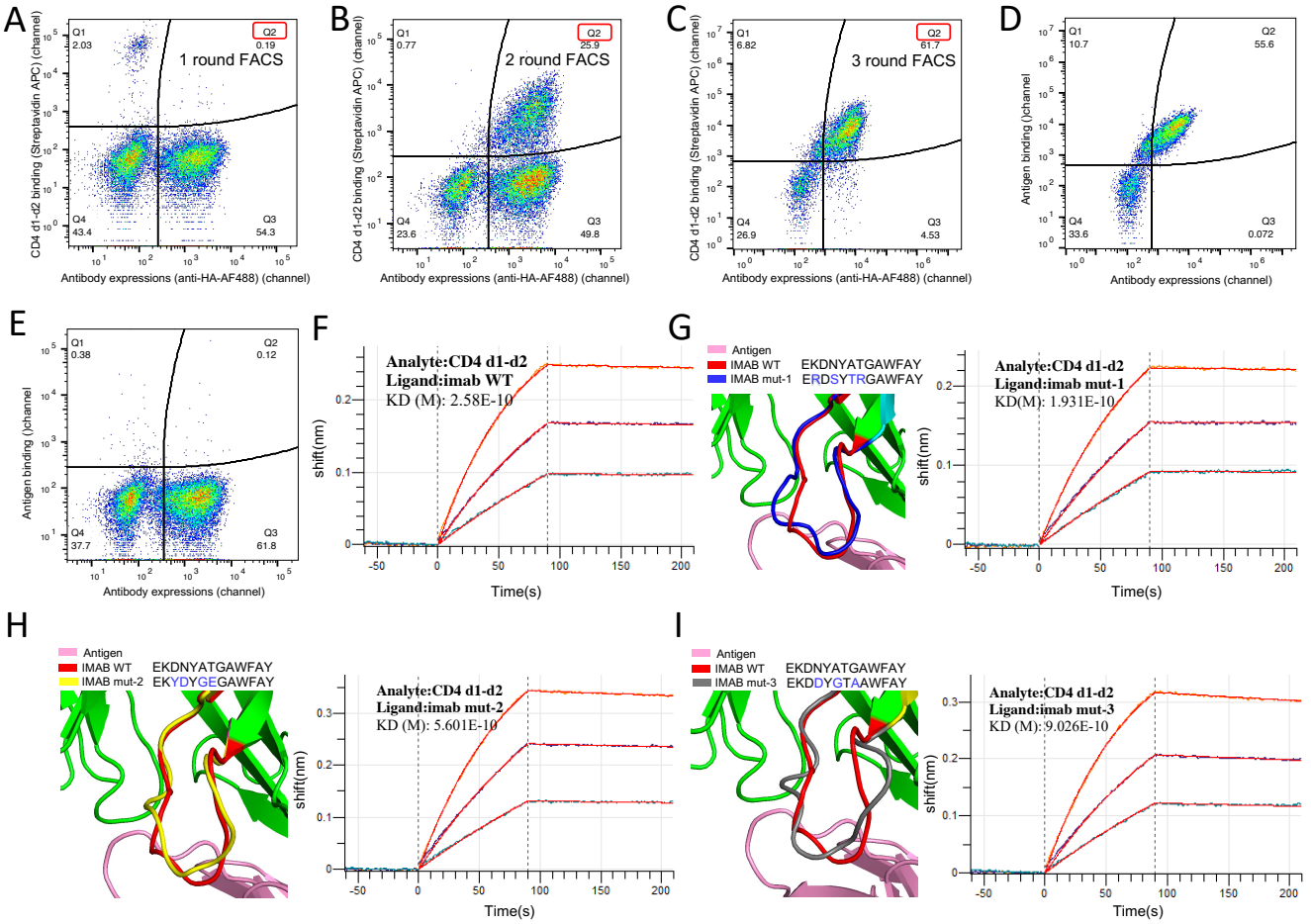
4.1 KEY RESOURCES TABLE

4.2 RESOURCE AVAILABILITY

4.2.1 Data and code availability

- The antibody dataset we used could be downloaded at `https://opig.stats.ox.ac.uk/webapps/sabdab-sabpred/sabdab`. And its sabdab summary file can be found in our GitHub repository(see key resources table for details).

- All original code has been open-sourced on GitHub. For details, please see the key resources table.



J

Virus name	Subtype	Tier	imab-WT	imab-mut-1	imab-mut-2	imab-mut-3
TRO_11	B	2	0.017	0.017	0.014	0.010
398_F1_F6_20	A1	2	0.010	0.010	0.010	0.012
CNE8	01_AE	2 or 3	0.013	0.013	0.013	0.013
X2278_C2_B6	B	2	0.013	0.018	0.007	0.012
BJOX002000_03_2	07_BC	2	0.022	0.033	0.019	0.013
25710_2_43	C	1B or 2	0.005	0.008	0.021	0.011
CNE55	01_AE	2	0.011	0.027	0.026	0.024
CNE4	2	B	0.024	0.015	0.006	0.011
CNE14	1B	B	0.017	0.019	0.013	0.010
SF162	1A	B	0.028	0.030	0.025	0.028
JRFL	2	B	0.089	0.055	0.156	0.058

IC_{50}(μg/ml)
0 .001
0.01
7.5

Figure 6. Performance for designed HIV-1 antibody variants. (A)-(C) Enrichment progress for designed antibody binding to CD4 through three rounds FACS. Yeast displayed single clones of designed antibodies Ibalizumab and FZD5 antibody 2919 acted as positive control and negative control, respectively. A red box is used to highlight the Q2 area and select its sequence for filtering.
(D, E) Binding kinetics of Ibalizumab and FZD5. **(F)-(I)** Binding kinetics of Ibalizumab and the top three isolated mAbs imab mut-1, imab mut-2 and imab mut-3 with CD4 d1-d2 measured by Biolayer Interferometry(BLI). The predicted structure of imab-mut1, imab-mut2, and imab-mut3 binding to antigen CD4 also showed in (G)-(I). **(J)** IC50 values of mutated antibodies and imab-WT against pseudotyped HIV-1 strains. Results were calculated from three independent experiments.

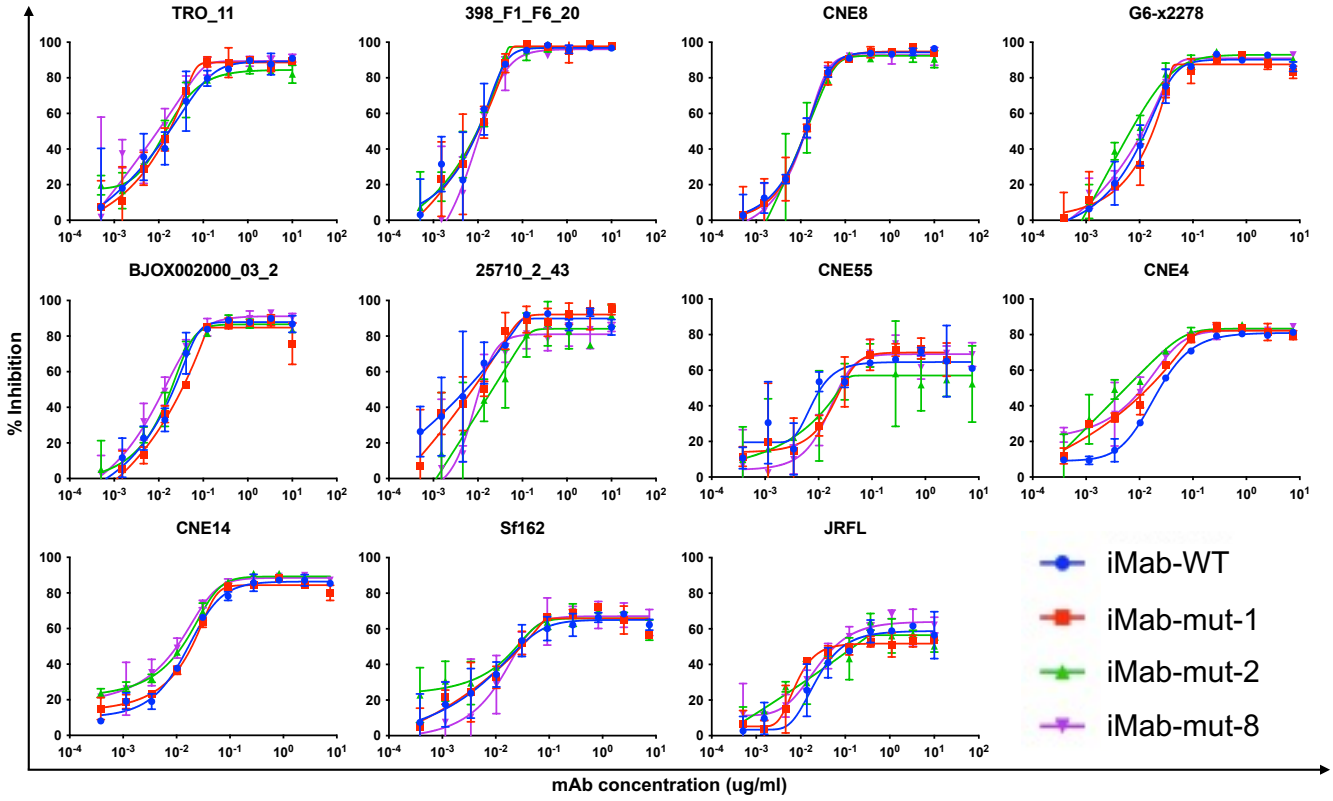


Figure 7. Neutralization curves of the generated antibodies and wild-type Ibalizumab against pseudotyped HIV-1 strains.

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited Data		
The Structural Antibody Database (SAbDab)	https://opig.stats.ox.ac.uk/webapps/sabdab-sabpred/sabdab	N/A
iMab sampling results	This Paper	https://doi.org/10.5281/zenodo.11228073
Software and Algorithms		
PyTorch	PyTorch	https://pytorch.org/
dyMEAN	Kong et al.[8]	https://github.com/THUNLP-MT/dyMEAN
RefineGNN	Jing et al.[9]	https://github.com/wengong-jin/RefineGNN
Diffab	Luo et al.[11]	https://github.com/luost26/diffab
HERN	Jing et al.[15]	https://github.com/wengong-jin/abdockgen
PyRosetta	PyRosetta	https://www.pyrosetta.org/
Rectified Flow	Liu et al.[19]	https://github.com/gnobitab/RectifiedFlow
Code for analysis	This paper	https://github.com/nohandsomewujun/FlowDesign

4.3 METHOD DETAILS

In this section, we have explained how to download and run the code for the baselines in Subsection 4.3.1. We describe the formalization of the antibody CDRs design task from the perspective of the Transport Mapping problem in Subsection 4.3.2. Next, in Subsection 4.3.3 and Subsection 4.3.4, we present the method for interpolation of the antibody as well as the loss function for training, respectively. Subsection 4.3.5 outlines the overall architecture of our model, and Subsection 4.3.6, Subsection 4.3.7 demonstrate the details of the data preparation and the prior distribution for training. Subsection 4.3.8 provides an overall description of the entire process of model training and inference, along with the selection of hyper-parameters.

4.3.1 Baselines

When evaluating performance, we selected baselines such as Diffab[11], RefineGNN[9], dyMEAN[8], and HERN[15]. Diffab is a diffusion-based generative model with approximately 4 million parameters, capable of simultaneously sampling and generating a large number of CDR structures and sequences. RefineGNN is an iterative refinement graph neural network for antibody sequence-structure co-design with approximately 6 million parameters. dyMEAN is an end-to-end full-atom model using an adaptive multi-channel equivariant encoder to update both 1D sequences and 3D structures with approximately 2 million parameters. HERN is used to design sequences of CDR via hierarchical equivariant refinement with approximately 7 million parameters. We used the InterfaceAnalyzer-Mover in PyRosetta to calculate binding energy. The weight setting PyRosetta used is REF2015.

The binding energy is a weighted sum of various Rosetta energy terms and can be referenced in the paper.[25] The corresponding code was downloaded from their respective GitHub repositories(see key resources table for details), and we re-partitioned the dataset(see data preparation for details) to retrain these models. In testing generative models like Diffab, we sample the same number of antibodies as the FlowDesign algorithm. For non-generative models, we generate one result for each antibody in the test set.

4.3.2 Transport Mapping Problem in Antibody CDRs' design

We begin by presenting an overview of the traditional Transport Mapping Problem:

Definition 4.1 (Transport Mapping)**.** Given the empirical observations of two distributions $X_0 \sim \pi_0$ and $X_1 \sim \pi_1$ on $\mathbb{R}^d$, we need to find a transport map T: $\mathbb{R}^d \to \mathbb{R}^d$, such that $Z_1 := T(Z_0) \sim \pi_1$ when $Z_0 \sim \pi_0$, that is, (Z_0, Z_1) is a coupling of π_0 and π_1[19].

The design of antibody CDRs can be similarly conceptualized as a Transport Mapping problem. First, we need to give the computational representation of antibody-antigen complexes as below. Following Diffab[11], an antibody-antigen complex is represented by the amino acid type $s_i \in \{\text{ACDEFGHIKLMNPQRSTVWY}\}$, the C_α atom coordinate $\boldsymbol{x_i} \in \mathbb{R}^3$, and the backbone orientation $\boldsymbol{O_i} \in SO(3)$ of each residue, where $i = 1...N$ for an antibody-antigen complex containing N residues. In addition, each s_i is encoded as a one-hot vector $\boldsymbol{s_i}$ as the input of the model. With the indexes of the residues in the CDR to design from $l+1$ to $l+m$, we denote the CDR as $\mathcal{R} = \{\boldsymbol{s}_i, \boldsymbol{x}_i, \boldsymbol{O}_i | i = l+1, ..., l+m\}$ and the rest of the antibody-antigen complex, which is given as the input, as $\mathcal{C} = \{\boldsymbol{s}_j, \boldsymbol{x}_j, \boldsymbol{O}_j | j \in \{1, ..., N\} \backslash \{l+1, ..., l+m\}\}$. The initial state of the CDRs, which is sampled from a specific distribution, is denoted as $\mathcal{T} = \{\boldsymbol{T}_{\boldsymbol{s}_i}, \boldsymbol{T}_{\boldsymbol{x}_i}, \boldsymbol{T}_{\boldsymbol{O}_i} | i = l+1, ..., l+m\}$. Now we are ready to formalize antibody CDR design as a Transport Mapping problem. Given $\mathcal{T} \sim \pi_{0|\mathcal{C}}$ and $\mathcal{R} \sim \pi_{1|\mathcal{C}}$, we aim to find a transport map T such that $T(\mathcal{T}) \sim \pi_{1|\mathcal{C}}$, where $\mathcal{T} \sim \pi_{0|\mathcal{C}}$ and $\mathcal{C}$ serves as the contextual condition. Obviously, such T can help us to generate the target CDR based on the premise of the framework.

4.3.3 Method of interpolation for antibodies

To endow the model with the capability of fine-grained flow matching, the training process takes the interpolation between the initial distribution and the target distribution as the input, according to a timestamp t following a uniform distribution between 0 and 1. Below we describe the interpolation method for the amino acid type, the C_α coordinates, and the backbone orientations, respectively. The amino acid types at position i are denoted as $\boldsymbol{s}_i$ and $\boldsymbol{T}_{\boldsymbol{s}_i}$ for the ground truth CDR and the initial CDR, respectively, which are encoded as one-hot vectors. Therefore, calculating the interpolation simply requires computing the linear interpolation between the two vectors:

$$\boldsymbol{s}_i^t = t \cdot \boldsymbol{s}_i + (1-t) \cdot \boldsymbol{T}_{\boldsymbol{s}_i} \quad t \sim \mathcal{U}(0,1) \tag{1}$$

Similarly, the C_α atom coordinates are also linearly interpolated as:

$$\boldsymbol{x}_i^t = t \cdot \boldsymbol{x}_i + (1-t) \cdot \boldsymbol{T}_{\boldsymbol{x}_i} \quad t \sim \mathcal{U}(0,1) \tag{2}$$

The interpolation of orientations differs slightly from the previous two cases. In order to ensure a smooth and uniform interpolation, we need to convert the orientations $\boldsymbol{O_i}$ and $\boldsymbol{T}_{\boldsymbol{O}_i}$ into quaternion representations: $\boldsymbol{q}_i$ and $\boldsymbol{T}_{\boldsymbol{q}_i}$. We then perform interpolation on the sphere using the spherical linear interpolation method, known as SLERP[26]:

$$\boldsymbol{q}_i^t = \frac{\sin(t \cdot \theta)}{\sin(\theta)} \cdot \boldsymbol{q}_i + \frac{\sin((1-t) \cdot \theta)}{\sin(\theta)} \cdot \boldsymbol{T}_{\boldsymbol{q}_i} \quad t \sim \mathcal{U}(0,1), \theta = arccos(\boldsymbol{q}_i \cdot \boldsymbol{T}_{\boldsymbol{q}_i}) \tag{3}$$

Therefore, for the $\mathcal{R} = \{\boldsymbol{s}_i, \boldsymbol{x}_i, \boldsymbol{O}_i | i = l+1, ..., l+m\}$ in the dataset and a sampled t, following the method described above, we can obtain its interpolated representation, which is denoted as $\mathcal{R}^t = \{\boldsymbol{s}_i^t, \boldsymbol{x}_i^t, \boldsymbol{O}_i^t | i = l+1, ..., l+m\}$

4.3.4 Loss Function

The training loss comprises individual losses on the amino acid types, the C_α atom coordinates, and the orientations, which are calculated as follows. Regarding amino acid types, we directly employ the Mean Squared Error (MSE) on the vectorized representations:

$$L_{seq}^t = \mathbb{E}\left[\frac{1}{m}\sum_{k=1}^{m} MSE(\boldsymbol{s}_k - \boldsymbol{T}_{\boldsymbol{s}_k}, F(\mathcal{R}^t, \mathcal{C}))\right], \tag{4}$$

where $F(\mathcal{R}^t, \mathcal{C})$ denotes the predicted drift force on the vectorized representation of amino acid types. Correspondingly, for the C_α atom coordinates, due to their continuity, we also use the MSE for calculation:

$$L_{pos}^t = \mathbb{E}\left[\frac{1}{m}\sum_{k=1}^{m} MSE(\boldsymbol{x}_k - \boldsymbol{T}_{\boldsymbol{x}_k}, G(\mathcal{R}^t, \mathcal{C}))\right], \tag{5}$$

where $G(\mathcal{R}^t, \mathcal{C})$ denotes the predicted drift force on the coordinates. The loss for orientations comprises two types of loss functions. First, we convert the orientation matrices into quaternions and implement Mean Squared Loss on them as follows:

$$L_{qua}^t = \mathbb{E}\left[\frac{1}{m}\sum_{k=1}^{m} MSE(\boldsymbol{q}_k \wedge \boldsymbol{T}_{\boldsymbol{q}_k}^*, H(\mathcal{R}^t, \mathcal{C}))\right], \tag{6}$$

where the $\wedge$ symbol indicates the Grassmann product between two quaternions, and the $*$ symbol denotes the inverse of a quaternion. Then we exploit the loss function directly implemented on $SO(3)$ the orientation matrices[11]:

$$L_{ori}^t = \mathbb{E}\left[\frac{1}{m}\sum_{k=1}^{m} \|\boldsymbol{O}_k^T \cdot (\boldsymbol{T}_{\boldsymbol{O}_k} \cdot Q(H(\mathcal{R}^t, \mathcal{C})) - \boldsymbol{I}\|\right], \tag{7}$$

where $F(\cdot)$, $G(\cdot)$, and $H(\cdot)$ are neural networks responsible for predicting the drift force that maps the initial distribution to the target distribution. Additionally, $Q(\cdot)$ is the function mapping the quaternion representation to the $SO(3)$ representation. By summing Eq.(4), (5), (6) and (7) and taking the expectation w.r.t t, we obtain the final training objective function:

$$L = \mathbb{E}_{t \sim Uniform(0,1)}\left[L_{seq}^t + L_{pos}^t + \alpha L_{qua}^t + \beta L_{ori}^t\right] \tag{8}$$

Here, both α and β range from 0 to 1, with their sum equaling one, balancing the two types of losses on the orientations.

4.3.5 Parameterization with Neural Network

In this section, we briefly introduce the overall architecture of the network. The primary task of our network is to encode the input complex and predict the drift force that maps the initial distribution of amino acid types (F), C_α atom coordinates (G), and orientations (H) to the target distribution.

The flow matching network incorporates a shared encoder for residue-level and pairwise representations, as well as three Multiple Layer Perceptron (MLP) heads for prediction of the drift forces on amino acid types, C_α atom coordinates, and orientations.

The residue-level input features include the amino acid type, all-atom coordinates, and side-chain torsional angles of each residue, which are processed into a vector e_i by a three-layer MLP with ReLU activation. The pairwise input features consist of the Euclidean distances and dihedral angles between each pair of residues, which are similarly encoded into vectors z_{ij} with another three-layer MLP. Both features are further concatenated with the embeddings of the flow matching time (t $\sim U(0,1)$) derived from two linear layers. Then, both e_i and z_{ij} are fed into an IPA module[27], which is an orientation-aware and roto-translation invariant network, to obtain the contextual hidden representations h_i of each residue.

Next, three separate MLP heads operate on the hidden representations h_i to predict the drift forces for amino acid types, C_α atom coordinates, and orientations. For the probability distribution of amino acid types, a three-layer MLP with ReLU activation is utilized, which outputs 20-dimensional vectors for each residue, indicating the expected drift force of 20 canonical types of amino acids. For the C_α atom coordinates, a similar MLP is used to predict the three-dimensional scaled change in the local coordinate system, which is further projected to the global coordinate system with the interpolated orientation matrix:

$$G(R^t, C) = (O_i)^t MLP_G(h_i) \tag{9}$$

Such local-to-global transformations ensure the roto-translation equivariance of the predicted drift force on coordinates. For the residue orientations, we adopted the quaternion method[28] which used four scalers to represent an orthogonal matrix. Therefore, an MLP with a four-dimensional output is utilized to predict the rotational drift force.

4.3.6 Data Preparation

Our model is trained on SAbDab dataset[29], which contains 13279 antibody-antigen complexes. We first clean up the data by discarding antibodies targeting non-protein antigens and structures with a resolution worse than 4Å. Next, to split the data for training, validation, and testing, we cluster all the antibodies in the dataset according to CDR-H3 sequences at 50% sequence identity using MMseqs2[24]. The test set comprises 20 non-redundant antibody-antigen complexes from different clusters. The complexes not in the same clusters as the test complexes are then partitioned into the training and the validation sets at a ratio of 20:1. In the training set, the average lengths of CDRs are as follows: CDR H1 is approximately 7.2, CDR H2 is 5.9, CDR H3 is 12.3, CDR L1 is 12.5, CDR L2 is 7.0, and CDR L3 is 9.2. In the test set, the average lengths are: CDR H1 is approximately 7.0, CDR H2 is 5.8, CDR H3 is 11.3, CDR L1 is 11.9, CDR L2 is 7.0, and CDR L3 is 9.2.

4.3.7 Prior Distributions

Below we provide the implementations of the prior distributions tested in our experiments (see Figure 2). We followed the dataset splitting mentioned above for the distribution generated by the dyMEAN[8] model. We fixed the sequences in the training data, excluding the antibody CDRs, and retrained the dyMEAN model. This model was then used to generate the initial CDRs for each complex in the dataset. For the random distribution, we directly leverage standard Gaussian for the vectorized representation of amino acid types. Regarding C_α atom coordinates, we rely on a standard Gaussian recentered at the centroid of the C_α atom coordinates from regions other than the CDR to be generated. As for the orientations, we initialize with $\mathcal{IG}_{SO(3)}$, the isotropic Gaussian distribution on SO(3) parameterized by a mean rotation and a scalar variance[30–32]. In terms of the distribution generated by the kinematic-closure-based method KIC, we begin by discarding

all structural information for the CDRs and using PyRosetta[33] to sample ϕ and ψ angles of the residues within CDR on the Ramachandran plot[34]. Subsequently, we solve the kinematic equations to achieve closure of the entire structure and select the lowest energy structures to serve as our structural prior. During this process, we set our closure attempt to 50,000, established a minimum solution constraint of 1 for the kinematic equations, and filtered out all structures that experienced collisions. For each antibody, we generated 100 corresponding structures to serve as its prior distribution for training and inference. To generate the sequence prior distribution produced by ESM2, we mask all residues in the antibody CDRs, and then utilize the esm2_t30_150M_UR50D model to predict the sequences for the CDR regions.

4.3.8 Model Training and Inference

In this section, we provide the implementation details for training and inference. During the training process, for the initialization of the distribution, we combine the dyMEAN distribution and the random distribution mentioned previously in Data Preparation by randomly selecting one method based on a probability p. Note that the probability p serves as a hyperparameter and can be adjusted to suit various requirements. To ensure the model does not get trapped in a local optimum and to guarantee the diversity of the initial distribution, we add a Gaussian noise when initializing the distribution with the dyMEAN model. Otherwise, the initial distribution would solidify into a few fixed samples, which would not be conducive to the generalization performance of the model. For α and β in the orientation loss, we set α to 1 and β to 0 for the initial 200,000 training iterations, and adjust α to 0 and β to 1 for the final 5,000 iterations. We provide further details in key resources table. We adopt Adam[35] optimizer for model optimization. Algorithm 1 below presents the pseudocode of the overall training process of our model.

Algorithm 1 Training

for $i = 1$ **to** `MAX_ITERATION` **do**
  $A_1 \leftarrow \{\mathcal{R}, \mathcal{C}\} \sim$ Training set
  Initialize a complex $A_0 = \{\mathcal{T}, \mathcal{C}\}$
  $t \sim \mathcal{U}(0, 1)$
  Calculate the interpolate representation $A_t = \{\mathcal{R}^t, \mathcal{C}\}$
  Choose α and β according to i
  Take gradient descent step on:
    $L^t_{\text{seq}} + L^t_{\text{pos}} + \alpha L^t_{\text{qua}} + \beta L^t_{\text{ori}}$
end for

Next, we illustrate the details of the inference process. The training objective fundamentally aims at finding a transformation within a high-dimensional space that maps our initialized complex A_0 directly to the real antibody A_1. While it is feasible to achieve this transformation in a single step, such a direct transformation will likely lead to inferior performance in practical applications. Therefore, we employ the integral method, which decomposes the transformation into finer steps to enhance the accuracy. Specifically, we rescale the predicted drift force at each step by $\frac{1}{T}$ and repeat the prediction-transformation cycle for T times. For the amino acid types and the C_α atom coordinates, we can simply multiply the output of the neural network with $\frac{1}{T}$ for transformations at each step.

For orientations, we explore two strategies during inference. The first strategy directly converts the quaternion predicted by our model into the rotational matrix, which is then applied to the input

orientation matrix via right multiplication. The second strategy scales the angle of the rotation θ by $\frac{1}{T}$ before the conversion of the quaternion. Our experiments demonstrate that the former method yields better results. Algorithm 2 below presents the pseudocode of the overall inference process of our model.

Algorithm 2 Inference

Given: the framework $\mathcal{C}$
Initialize: initialize the targeting CDR with $\mathcal{T} = \{\boldsymbol{T_s}, \boldsymbol{T_x}, \boldsymbol{T_O}\}$
$\mathcal{R}_{\text{pre}} = \{\boldsymbol{s}_{\text{pre}} = \boldsymbol{T_s}, \boldsymbol{x}_{\text{pre}} = \boldsymbol{T_x}, \boldsymbol{O}_{\text{pre}} = \boldsymbol{T_O}\} = \mathcal{T}$
for $i = 1, \ldots, T$ **do**
    $\boldsymbol{s}_{\text{update}} = \boldsymbol{s}_{\text{pre}} + \frac{1}{T} \cdot F(\mathcal{R}_{\text{pre}}, \mathcal{C})$
    $\boldsymbol{x}_{\text{update}} = \boldsymbol{x}_{\text{pre}} + \frac{1}{T} \cdot G(\mathcal{R}_{\text{pre}}, \mathcal{C})$
    $\boldsymbol{O}_{\text{update}} = \boldsymbol{O}_{\text{pre}} \cdot Q(H(\mathcal{R}_{\text{pre}}, \mathcal{C}))$
    $\mathcal{R}_{\text{pre}} = \{\boldsymbol{s}_{\text{pre}} = \boldsymbol{s}_{\text{update}}, \boldsymbol{x}_{\text{pre}} = \boldsymbol{x}_{\text{update}}, \boldsymbol{O}_{\text{pre}} = \boldsymbol{O}_{\text{update}}\}$
end for
return $A_{\text{pre}} = \{\mathcal{R}_{\text{pre}}, \mathcal{C}\}$

4.3.9 Experimental procedure

In order to assess the potential of FlowDesign in generating high-affinity antibodies via cycles of yeast culture and fluorescence-activated cell sorting (FACS), our objective was to evolve the antibody Ibalizumab. The designed Ibalizumab HCDR3 library was codon-optimized for expression in yeast and cloned into pCTCON2 for galactose induction and surface expression in S.cerevisiae using the Aga1p-Aga2p protein display system. The plasmid library was then transformed into the EBY100 strain using the Frozen Yeast Transformation Kit (Coolaber, SK2400). The constructed library was further cultured overnight in SD-CAA medium, followed by induction in SG-CAA medium. After induction, yeast cells were incubated with lab-made antigen CD4 d1-d2 for 1 hour and subsequently washed with PBS buffer. After incubation with fluorescently labeled antibodies, three consecutive rounds of sorting were performed on a BD Aria cell sorter. For the first round of FACS, 5×10^7 induced yeast cells were centrifuged and washed with PBS buffer. And yeast cells were then incubated with 1 μM biotin-conjugated CD4 d1-d4. After washing off unbound antigens, cells were then stained with 1:200 diluted anti-HA AF488 (Cell Signaling Technology cat. no. 2350) and Streptavidin APC (Miltenyi Biotec cat. no. 130-106-791) antibody for 1 hour at 4 °C. These cells were then centrifuged, resuspended in 3 ml of PBS buffer, and sorted. Sorting gates were defined by utilizing control cells displaying an antibody and stained with secondary antibodies, albeit without prior antigen incubation. The collection window is designed to isolate cells that secreted and displayed a functional antibody. In total, around 23,000 cells from 15,000,000 were collected and expanded. The second round of FACS was performed with similar conditions to the first; however, the antigen was changed to lab-made CD4 d1-d2, and the concentration of CD4 was reduced to 500 nM. For the second round, 3,000,000 cells were collected. For the last round of FACS, concentration of antigen was decreased to 200nM, while other conditions remained the same. In total, 3,000,000 cells were collected and plated on SD-CAA medium to isolate single clones. Forty-eight colonies were picked, cultured, induced and finally verified for binding with the antigen CD4 d1-d2. After three rounds of sorting, yeast cells were expanded to isolate yeast plasmids by using Yeast Plasmid Miniprep Kit (Zoman Biotechnology, ZP107). Yeast plasmids were then used as templates to amplify HCDR3 fragments to perform next-generation sequencing.

After NGS analysis, the top 10 mutants were selected and expressed for characterization.

Antibodies targeting CD4 d1-d2 were expressed by cloning into pcDNA3.4 (biointron). For each antibody, 20mL of Expi293F cells (Thermo Fisher, A14527) were transfected with 20ug of plasmids. After one day, cells were enhanced with ExpiFectamine 293 Transfection Enhancer 1 and Enhance 2(Thermo Fisher, A14525). Cell supernatants were collected four days after transfection and each antibody was purified with 1mL rProtein A Beads(smart-life sciences, SA012100), which was then washed with 20 ml of PBS, eluted with 0.1M glycine (pH 3) and then neutralized with concentrated Tris buffer (pH 8). All antibodies were then dialyzed twice with PBS buffer.

Biolayer interferometry experiments for antibody binding assays were performed with an Octet R8 system (Sartorius). For antibody assays, Ibalizumab and its mutants were loaded onto protein A sensors (ForteBio, Pall LLC) at 10 μg/ml for 120 s. The baseline interference was then read for 60 s in KB buffer (PBS, 0.1% BSA, 0.02%Tween). After the baseline measurement was obtained, antibodies were associated at 31.3 nM for 90 s, followed by an association KB buffer for 120s.

Pseudoviruses bearing the 11 HIV-1 envelopes (TRO_11, 398_F1_F6_20, CNE8, X2278_C2_B6, BJOX002000_03_2, 25710_2_43, CNE55, CNE4, CNE14, SF162 and JRFL) were generated by co-transfecting HEK293T cells with Env expression vectors and the pNL4–3R-Eluciferase viral backbone plasmid as described previously[36]. Pseudovirus-containing supernatants were collected 48 or 72 hours post transfection and the viral titers were measured by luciferase activity in relative light units (RLU) (Bright-Lite Luciferase Assay System, Vazyme, China). The supernatants were aliquoted and stored at -80°C until further use. Neutralization assays were performed by adding approximately 1.5×10^4 GhostX4/R5 cells into 10 serial 1:3 dilutions of purified antibody starting from 7.5 μg/ml. The mixture was then dispensed into a 96-well plate in duplicate and incubated for 1 hour at 37°C. The pseudovirus supernatants were then added and the cultures were maintained at 37°C for an additional 48h before luciferase activity was measured. Neutralizing activity was measured by the reduction in luciferase activity compared to the controls. The half-maximal inhibitory concentrations (IC50), the concentrations required to inhibit infection by 50% compared to the controls, were calculated using the dose-response-inhibition model with a 5-parameter Hill slope equation in GraphPad Prism 10(GraphPad Software, USA.).

References

1. Raybould, M. I. J. *et al.* Five computational developability guidelines for therapeutic antibody profiling. *Proceedings National Academy Sciences* 4025–4030, DOI: 10.1073/pnas.1810576116 (2019).
2. Forthal, D. N. Functions of antibodies. *Microbiology spectrum* **2**, 2–4 (2014).
3. Pantazes, R. J. & Maranas, C. D. Optcdr: a general computational method for the design of antibody complementarity determining regions for targeted epitope binding. *Protein Engineering Design Selection* **23**, 849–858, DOI: 10.1093/protein/gzq061 (2010).
4. Lapidoth, G. D. *et al.* Abdesign: An algorithm for combinatorial backbone design guided by natural conformations and sequences. *Proteins: Structure, Function, Bioinformatics* 1385–1406, DOI: 10.1002/prot.24779 (2015).
5. Adolf-Bryfogle, J. *et al.* Rosettaantibodydesign (rabd): A general framework for computational antibody design (2017). DOI: 10.1101/183350.
6. Warszawski, S. *et al.* Optimizing antibody affinity and stability by the automated design of the variable light-heavy chain interfaces. *PLoS computational biology* **15**, e1007207 (2019).

7. Kong, X., Huang, W. & Liu, Y. Conditional antibody design as 3d equivariant graph translation. *arXiv preprint arXiv:2208.06073* (2022).

8. Kong, X., Huang, W. & Liu, Y. End-to-end full-atom antibody design. *arXiv preprint arXiv:2302.00203* (2023).

9. Jin, W., Wohlwend, J., Barzilay, R. & Jaakkola, T. Iterative refinement graph neural network for antibody sequence-structure co-design. *arXiv preprint arXiv:2110.04624* (2021).

10. Jin, W., Barzilay, R. & Jaakkola, T. Antibody-antigen docking and design via hierarchical equivariant refinement. (2022).

11. Luo, S. *et al.* Antigen-specific antibody design and optimization with diffusion-based generative models. .

12. Watson, J. L. *et al.* De novo design of protein structure and function with rfdiffusion. *Nature* **620**, 1089–1100 (2023).

13. Lin, Z. *et al.* Language models of protein sequences at the scale of evolution enable accurate structure prediction. *BioRxiv* **2022**, 500902 (2022).

14. Nijkamp, E., Ruffolo, J. A., Weinstein, E. N., Naik, N. & Madani, A. Progen2: exploring the boundaries of protein language models. *Cell Systems* **14**, 968–978 (2023).

15. Jing, H. *et al.* Accurate prediction of antibody function and structure using bio-inspired antibody language model. *bioRxiv* 2023–08 (2023).

16. Shuai, R. W., Ruffolo, J. A. & Gray, J. J. Iglm: Infilling language modeling for antibody sequence design. *Cell Systems* **14**, 979–989 (2023).

17. Olsen, T. H., Moal, I. H. & Deane, C. M. Ablang: an antibody language model for completing antibody sequences. *Bioinformatics Advances* **2**, vbac046 (2022).

18. Bennett, N. R. *et al.* Atomically accurate de novo design of single-domain antibodies. *bioRxiv* 2024–03 (2024).

19. Liu, X., Gong, C. & Liu, Q. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003* (2022).

20. Jacobson, J. M. *et al.* Safety, pharmacokinetics, and antiretroviral activity of multiple doses of ibalizumab (formerly tnx-355), an anti-cd4 monoclonal antibody, in human immunodeficiency virus type 1-infected adults. *Antimicrobial agents chemotherapy* **53**, 450–457 (2009).

21. Abramson, J. *et al.* Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature* 1–3 (2024).

22. Mandell, D. J., Coutsias, E. A. & Kortemme, T. Sub-angstrom accuracy in protein loop reconstruction by robotics-inspired conformational sampling. *Nature methods* **6**, 551–552 (2009).

23. Stein, A. & Kortemme, T. Improvements to robotics-inspired conformational sampling in rosetta. *PloS one* **8**, e63090 (2013).

24. Steinegger, M. & Söding, J. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology* **35**, 1026–1028 (2017).

25. Alford, R. F. *et al.* The rosetta all-atom energy function for macromolecular modeling and design. *Journal chemical theory computation* **13**, 3031–3048 (2017).

26. Shoemake, K. Animating rotation with quaternion curves. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, 245–254 (1985).

27. Jumper, J. *et al.* Highly accurate protein structure prediction with alphafold. *Nature* **596**, 583–589 (2021).

28. Gallier, J. & Xu, D. Computing exponentials of skew-symmetric matrices and logarithms of orthogonal matrices. *International Journal Robotics Automation* **18**, 10–20 (2003).

29. Dunbar, J. *et al.* SAbDab: the structural antibody database. *Nucleic Acids Research* **42**, D1140–D1146, DOI: 10.1093/nar/gkt1043 (2013). https://academic.oup.com/nar/article-pdf/42/D1/D1140/3538157/gkt1043.pdf.

30. Leach, A., Schmon, S., Degiacomi, M. & Willcocks, C. Denoising diffusion probabilistic models on so(3) for rotational alignment. .

31. Matthies, S., Muller, J. & Vinel, G. W. On the normal distribution in the orientation space. *Textures Microstructures* **10**, 77–96, DOI: 10.1155/tsm.10.77 (1988).

32. Nikolayev, D. I. & Savyolov, T. I. Normal distribution on the rotation group so(3). *Textures Microstructures* **29**, 201–233, DOI: 10.1155/tsm.29.201 (1997).

33. Chaudhury, S., Lyskov, S. & Gray, J. J. Pyrosetta: a script-based interface for implementing molecular modeling algorithms using rosetta. *Bioinformatics* **26**, 689–691 (2010).

34. Hollingsworth, S. A. & Karplus, P. A. A fresh look at the ramachandran plot and the occurrence of standard structures in proteins. (2010).

35. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

36. Li, M. *et al.* Human immunodeficiency virus type 1 env clones from acute and early subtype b infections for standardized assessments of vaccine-elicited neutralizing antibodies. *Journal Virology* **79**, 10108–10125, DOI: 10.1128/jvi.79.16.10108-10125.2005 (2005). https://journals.asm.org/doi/pdf/10.1128/jvi.79.16.10108-10125.2005.

37. Eastman, P. *et al.* Openmm 7: Rapid development of high performance algorithms for molecular dynamics. *PLoS computational biology* **13**, e1005659 (2017).

38. Schymkowitz, J. *et al.* The foldx web server: an online force field. *Nucleic acids research* **33**, W382–W388 (2005).

5 SUPPLEMENTAL INFORMATION

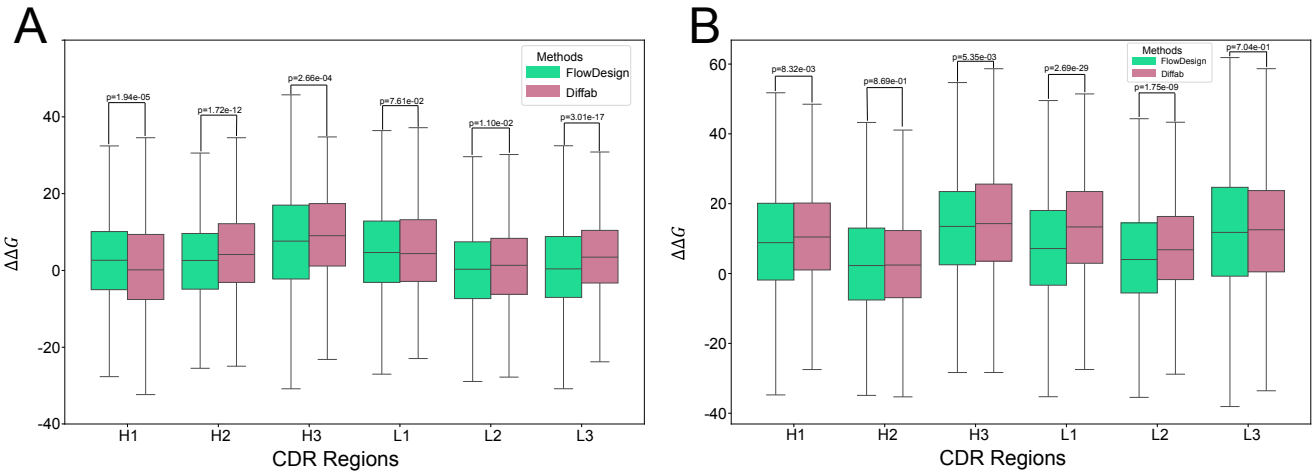


Figure S1. Comparison of $\Delta\Delta G$ for different side chain generation strategies and energy calculation methods on different CDRs. **(A)** $\Delta\Delta G$ comparison using OpenMM[37] for side-chain packing. **(B)** $\Delta\Delta G$ comparsion using FoldX[38] for binding energy calculation.

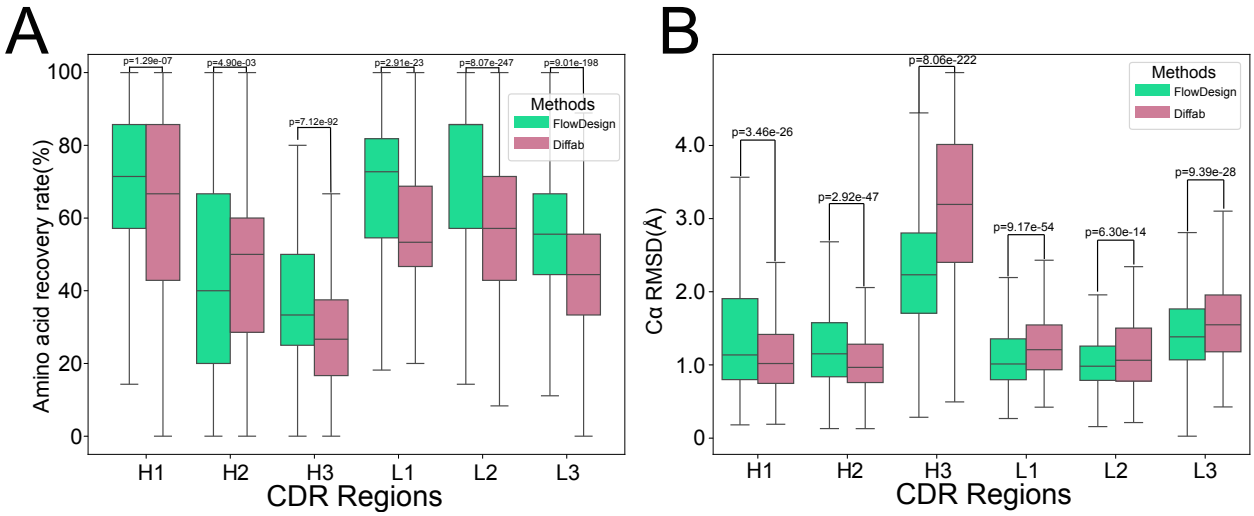


Figure S2. Comparison of performance between randomly initialized FlowDesign and Diffab on different CDRs. **(A)** Comparison of the distribution of AAR in generated conformations. **(B)** Comparison of the distribution of RMSD in generated conformations.

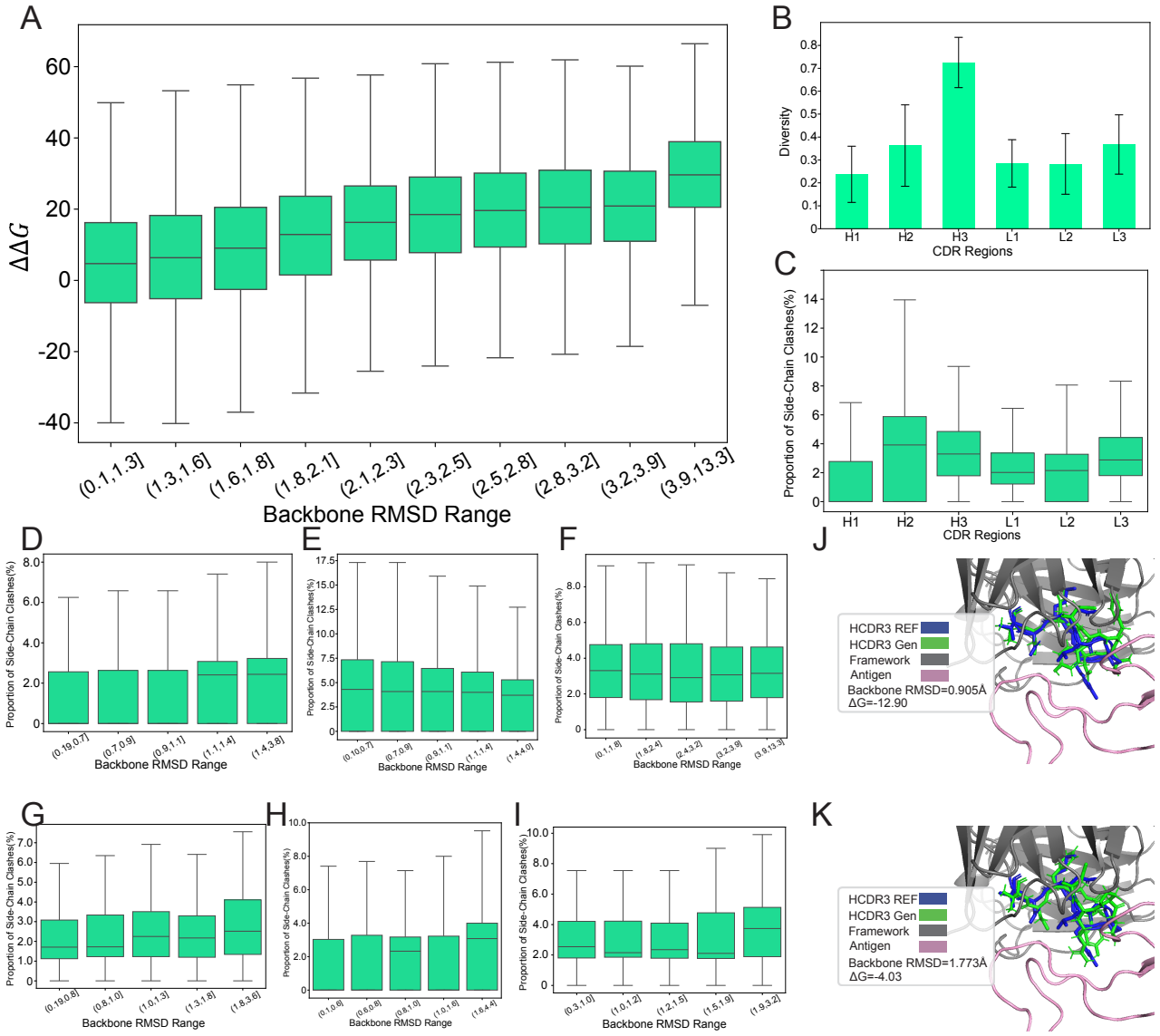


Figure S3. **(A)** The relationship between the RMSD of the generated conformations' backbone and energy. **(B)** The diversity of sequences in the generated conformations. **(C)** Proportion of side-chain clashes in different CDRs. **(D)-(I)** The relationship between the RMSD of the generated conformations' backbone and side-chain clashes in different CDRs. In order: CDR H1, H2, H3, L1, L2, L3. **(J)(K)** Visualization of generated conformations and their side-chains.

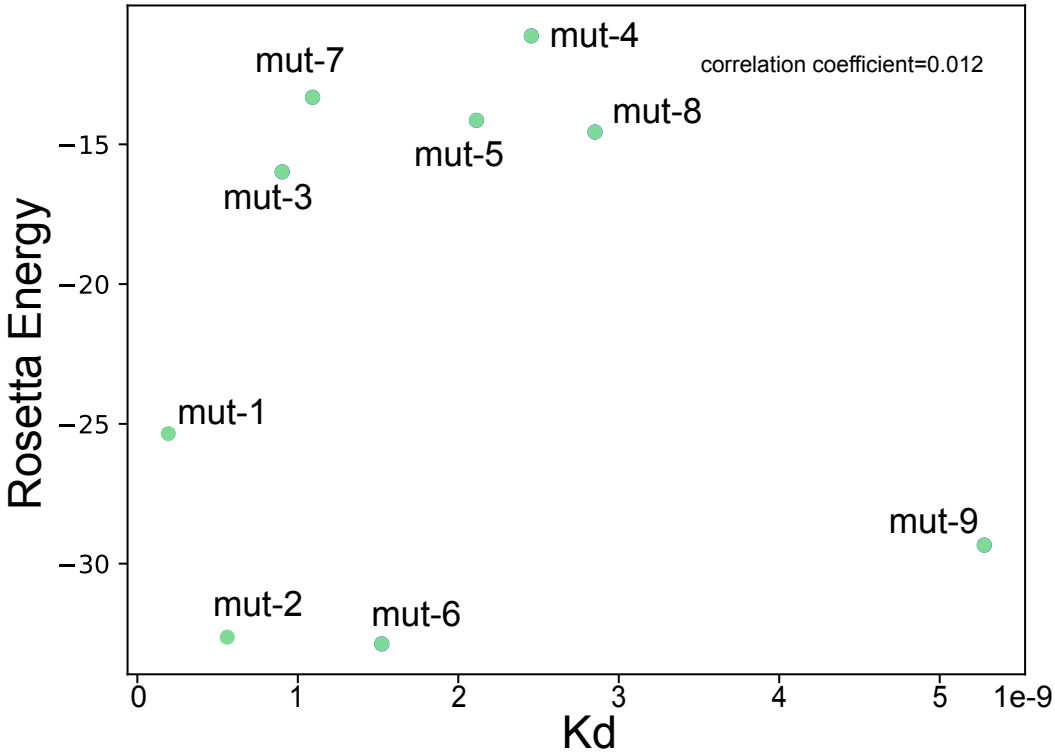


Figure S4. The relationship between the Kd of HIV mutant antibodies and their corresponding Rosetta energy.

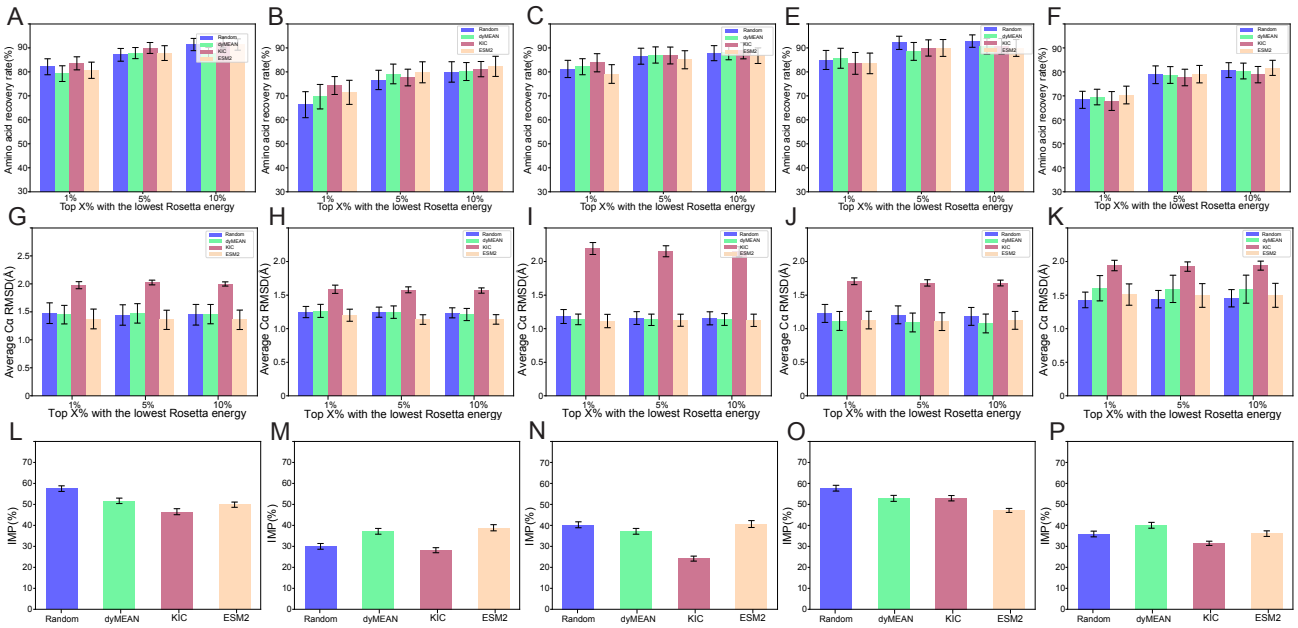


Figure S5. **(A)-(F)** Performance for amino acid recovery rates using different prior distributions in CDR H1, H2, L1, L2, and L3. **(G)-(K)** Performance for average C_α RMSD using different prior distributions in CDR H1, H2, L1, L2, and L3. **(L)-(P)** Performance for IMP using different prior distributions in CDR H1, H2, L1, L2, and L3.

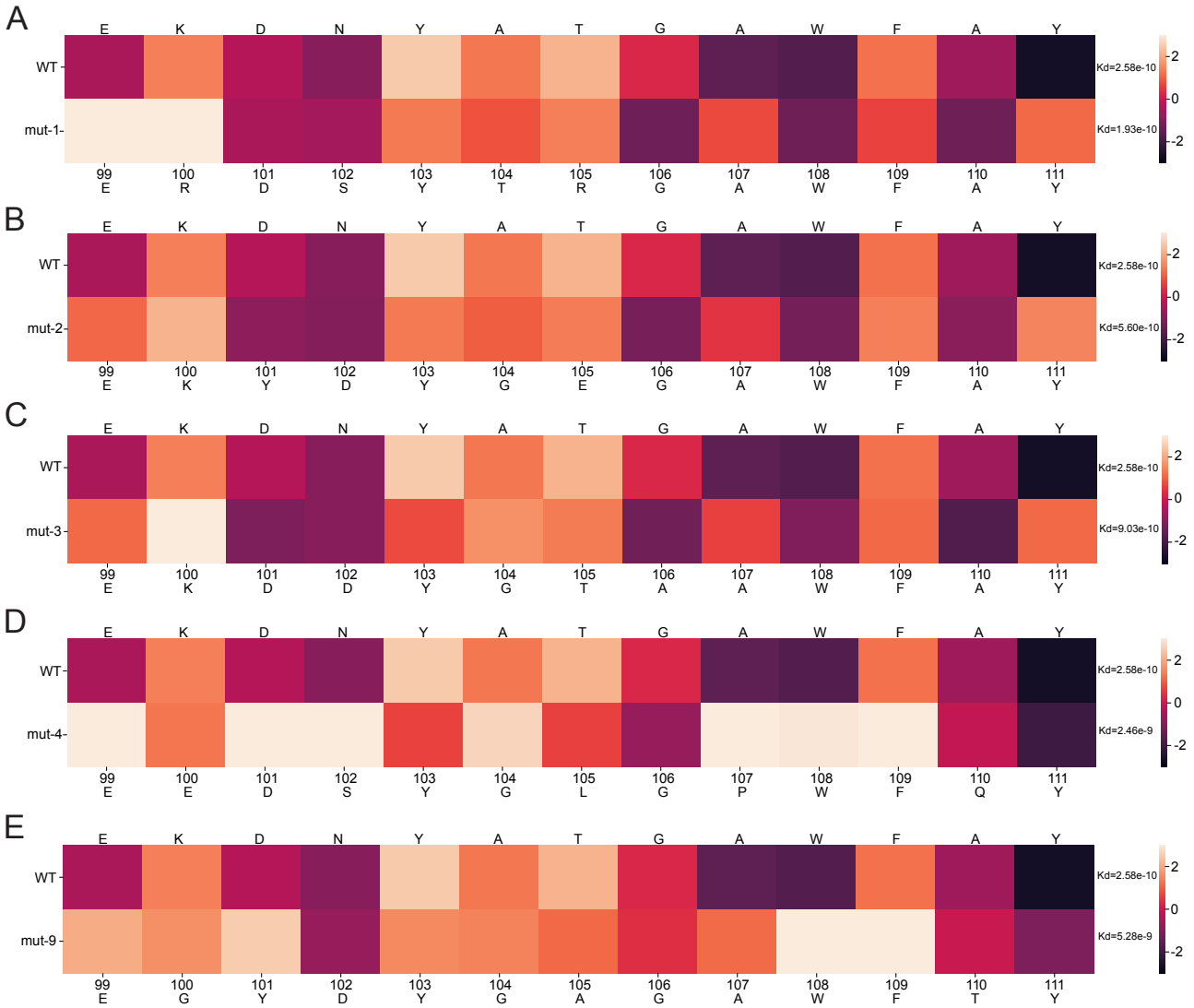


Figure S6. **(A)-(E)** Visualizations of the energy change for each residue in the CDRH3 of imab-mut 1, 2, 3, 4, and 9 relative to imab-WT, along with their corresponding Kd values.