End-to-End Full-Atom Antibody Design

Xiangzhe Kong [1 2] **Wenbing Huang** [3 4] **Yang Liu** [1 2]

Abstract

Antibody design is an essential yet challenging task in various domains like therapeutics and biology. There are two major defects in current learning-based methods: 1) tackling only a certain subtask of the whole antibody design pipeline, making them suboptimal or resource-intensive. 2) omitting either the framework regions or side chains, thus incapable of capturing the full-atom geometry. To address these pitfalls, we propose **dy**namic **M**ulti-channel **E**quivariant gr**A**ph **N**etwork (dyMEAN), an end-to-end full-atom model for E(3)-equivariant antibody design given the epitope and the incomplete sequence of the antibody. Specifically, we first explore *structural initialization* as a knowledgeable guess of the antibody structure and then propose *shadow paratope* to bridge the epitope-antibody connections. Both 1D sequences and 3D structures are updated via an *adaptive multi-channel equivariant encoder* that is able to process protein residues of variable sizes when considering full atoms. Finally, the updated antibody is docked to the epitope via the alignment of the shadow paratope. Experiments on epitope-binding CDR-H3 design, complex structure prediction, and affinity optimization demonstrate the superiority of our end-to-end framework and full-atom modeling.

1. Introduction

Antibodies are a family of Y-shaped proteins in immune systems that binds to pathogens, commonly called antigens, with specificity (Raybould et al., 2019). Antibody design for target epitopes on the antigen exhibits tremendous potential
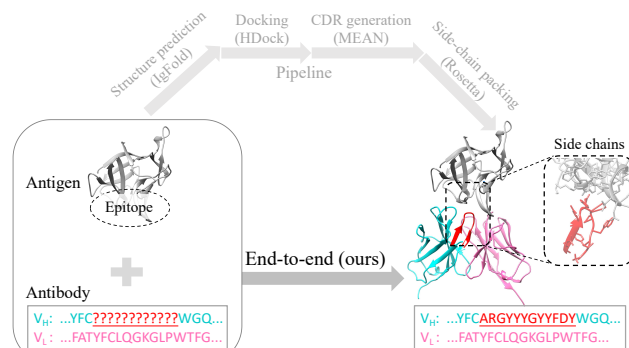

Figure 1. Our end-to-end full-atom antibody design. By contrast, current computational methods resort to the multi-stage solution: e.g., IgFold (Ruffolo & Gray, 2022) for structure prediction, HDock (Yan et al., 2020) for docking on the target epitope, MEAN (Kong et al., 2022) for binding CDR generation, and Rosetta (Alford et al., 2017) for side-chain packing.

and necessity in therapeutic and biological research (Tiller & Tessier, 2015; Almagro et al., 2018; Yuan et al., 2020). Nevertheless, the task is challenging because the *complementarity determining regions (CDRs)*, where the binding mainly occurs, are highly variant, and the underlying regularity of antigen-antibody interactions is arduous to unveil. The past decade has seen the application of traditional energy-based optimization (Li et al., 2014; Adolf-Bryfogle et al., 2018), learning-based language models on the 1D sequence (Liu et al., 2020; Saka et al., 2021), as well as recent deep generative methods to co-design the CDR sequences and 3D structures simultaneously, exhibiting appealing superiority over conventional sequence-based approaches. (Jin et al., 2021; Luo et al., 2022; Kong et al., 2022).

Despite the impressive progress, current computational models are still incapable of fulfilling the real need for antibody design. In most practical cases, we only know the 3D structure of antigen with the target epitope and the 1D incomplete sequence (without CDRs) of antibody. To address this ill-posed task, a potential computational pipeline includes: structure prediction (Ruffolo & Gray, 2022), antigen-antibody docking (Yan et al., 2020), binding CDR generation (Jin et al., 2022; Luo et al., 2022; Kong et al., 2022), and side-chain packing (Alford et al., 2017), as illustrated in Figure 1. Existing works can solve each local problem separately, but lacks the mastery of the global picture, making them suboptimal. Conducting wet-lab experiments, such as obtaining the antigen-antibody complex structure

---

[1]Dept. of Comp. Sci. & Tech., Institute for AI, BNRist Center, Tsinghua University [2]Institute for AI Industry Research (AIR), Tsinghua University [3]Gaoling School of Artificial Intelligence, Renmin University of China [4]Beijing Key Laboratory of Big Data Management and Analysis Methods, Beijing, China. Correspondence to: Wenbing Huang <hwenbing@126.com>, Yang Liu <liuyang2011@tsinghua.edu.cn>.

from cryo-electron microscopy, somehow bypasses this sub-optimality, yet is much more costly and laborious (Carter, 2006). Therefore, the deficiency of both computational pipelines and experimental methods poses an urgent need for a computational end-to-end solution.

Furthermore, the full-atom geometry is critical for depicting the interactions within the antigen-antibody complex (Foote & Winter, 1992; Jones & Thornton, 1996). Current works usually model the backbone atoms only (Jin et al., 2021; Kong et al., 2022), or simply consider the orientation of side chains (Luo et al., 2022). Although Jin et al. (2022) makes an initial attempt to incorporate all side-chain atoms into a hierarchical graph, it suffers from efficiency problems and is obliged to omit all other components of the antibody except CDR-H3 (see Appendix G), leading to incomplete context modeling and thus inaccurate design. Full-atom geometry of the entire antibody has a much larger scale and demands a computationally more efficient and effective model.

To address the above two issues, we propose **dy**namic **M**ulti-channel **E**quivaraint gr**A**ph **N**etwork (dyMEAN) as an end-to-end and full-atom solution. Compared to previous works (Luo et al., 2022; Kong et al., 2022), we directly tackle the end-to-end problem where only the epitope and the incomplete 1D sequence are known in advance (Figure 1), in contrast to previous multi-stage solutions. We explore knowledge-guided *structural initialization* based on conserved residues and propose *shadow paratope* to capture antigen-antibody interaction that is invariant to their initial orientations and positions. The 1D sequence and the 3D structure are updated iteratively via an *adaptive multi-channel message passing*, which favorably tolerates the variance in the number of channels (*i.e.*, atoms) in different residues, when considering full-atom geometry. We finally achieve epitope-antibody docking through the alignment of the shadow paratope. The network also conforms to E(3)-equivariance, which is a critical property exhibited in 3D biology (Kong et al., 2022). Experiments on epitope-binding CDR-H3 design, complex structure prediction, and affinity optimization demonstrate the superiority of our end-to-end framework and full-atom modeling.

2. Related Work

Antibody Design   Conventional computational methods commonly optimize sophisticated energy functions designed by domain experts (Li et al., 2014; Adolf-Bryfogle et al., 2018), or train language models on the 1D sequences (Liu et al., 2020; Saka et al., 2021; Akbar et al., 2022). Energy-based methods suffer from the insufficient expressive power of the statistical energy functions (MacKerell Jr et al., 2002; Leaver-Fay et al., 2011), and language models are suboptimal due to the lack of structural modeling. More recently, the community has witnessed the emergence of sequence-

structure co-design methods and their superiority over previous methods (Jin et al., 2021; 2022; Luo et al., 2022; Kong et al., 2022). However, they are limited to certain stages of pipeline-based antibody design. For example, Jin et al. (2021) generates the CDRs on a single chain, and Luo et al. (2022); Kong et al. (2022) fill in the CDRs given a docked complex, demanding hard-to-obtain prerequisites. Jin et al. (2022) attempts to generate and dock CDR-H3 simultaneously on the local binding interface. Nevertheless, it suffers from the inefficiency of the distance-based initialization, the hierarchical encoding, and the autoregressive refinement (see Appendix G), preventing it from scaling to the entire antibody. Distinct from the above works, we directly generate the complete complex given the epitope and the incomplete sequence in an end-to-end and full-atom manner.

Protein Docking   Generally, protein docking predicts the docked complex of two proteins given their unbound structures (Kozakov et al., 2017; Yan et al., 2020; Ganea et al., 2021). While they require the structure of both proteins in advance, our work simultaneously generates the structure of the antibody and docks it to the antigen. Another difference lies in the prior knowledge of the binding regions on the antigen and the antibody (*i.e.*, the epitope and the paratope). Only certain epitopes on the antigen constitute meaningful targets in therapeutics (Yuan et al., 2020), and the paratope mostly comes from CDRs, especially CDR-H3 (Kuroda et al., 2012). Therefore, antibody docking mainly focuses on the local binding interface, while many protein docking methods (e.g., EquiDock, Ganea et al., 2021) assume no prior knowledge of the epitope and the paratope, making them suboptimal in this situation.

Equivariant Graph Neural Networks   Equivariant graph neural networks are designed with the desired inductive bias that the results should not rely on the view of observation, namely E(3)-equivariance. With increasing availability of 3D data, abundant equivariant neural networks have emerged (Thomas et al., 2018; Gasteiger et al., 2020; Fuchs et al., 2020; Satorras et al., 2021). Our work is closely related to the multi-channel equivariant graph networks proposed by Kong et al. (2022), where each residue node has multiple coordinates (i.e., channels) referring to different atoms. We propose a more powerful version of multi-channel equivariant message passing, which is adaptive to the variable number of channels in full-atom modeling.
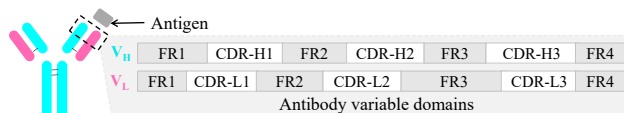
3. Notations and Definitions

Figure 2. Variable domains in the heavy/light chain (V_H / V_L).

A protein comprises one or more long chains of amino acid residues. An antibody is a Y-shaped symmetric protein
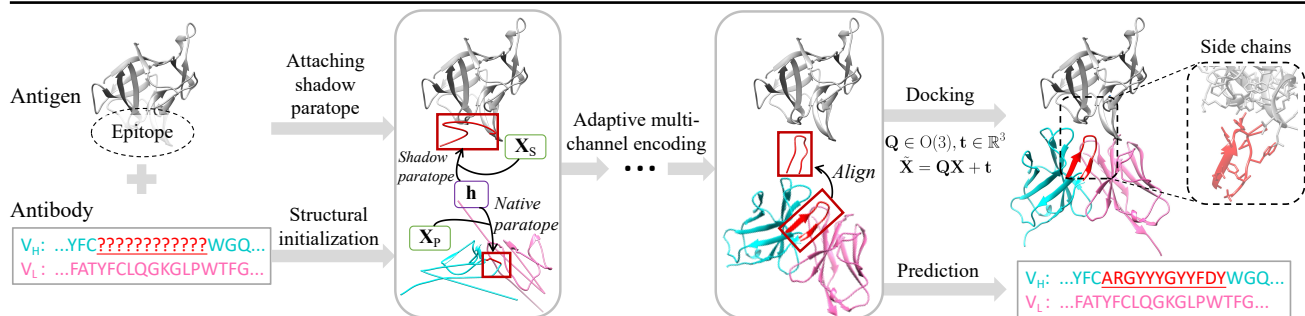
Figure 3. Overall architecture. **Structural initialization** (§ 4.1): obtaining the initial hidden vector $\boldsymbol{h}_i$ and coordinate matrix $\boldsymbol{X}_i$ for the antibody. **Attaching shadow paratope** (§ 4.2): attaching a clone of the paratope around the epitope, where $\boldsymbol{h}_i$ is shared, but the coordinates are private for the shadow paratope and the native one, *i.e.*, $\boldsymbol{X}_S$ v.s. $\boldsymbol{X}_P$. **Adaptive multi-channel encoding** (§ 4.3): updating $\boldsymbol{h}_i$ and $\boldsymbol{X}_i$ by multi-channel message passing, where the full-atom geometry is characterized. **Docking** (§ 4.4): Aligning the native paratope to the shadow paratope. **Prediction** (§ 4.4): outputting the amino acid type of each residue in the paratope.

with two identical sets of chains, as illustrated in Figure 2. Each set contains a heavy chain and a light chain, either of which consists of several constant domains and a variable domain. As their names suggest, the constant domains keep unchanged across different antibodies; while the variable domain varies to enable different binding specificity for different antigens, making it the main focus of antibody design. We denote the variable domains of the heavy chain and the light chain by V_H and V_L, respectively. The variable domain is further divided into alternating arrangements of four *framework regions (FRs)* and three *complementarity determining regions (CDRs)*. The binding regions of an antigen and an antibody are called an *epitope* and a *paratope*, separately. In this paper, the paratope refers to CDR-H3 in the heavy chain following Jin et al. (2022), since it is highly variable and dominates binding (MacCallum et al., 1996).

We describe the epitope of the antigen and the variable domains of the antibody as the graphs $\mathcal{G}_E(\mathcal{V}_E, \mathcal{E}_E)$ and $\mathcal{G}_A(\mathcal{V}_A, \mathcal{E}_A)$, where $\mathcal{V}_E$ and $\mathcal{V}_A$ refer to the vertices (*i.e.*, the residues), $\mathcal{E}_E$ and $\mathcal{E}_A$ are edges. Each residue v_i is represented by its amino acid type s_i and a multi-channel 3D coordinate matrix $\boldsymbol{X}_i \in \mathbb{R}^{3 \times c_i}$, where c_i denotes the channel size, *i.e.*, the number of atoms in v_i. Notably, previous studies (Kong et al., 2022) only consider backbone atoms for each residue, and the coordinate dimension is constant: $c_i = 4$. This paper models the full-atom geometry by further involving side chains, hence c_i is distinct for different residues. The edges are constructed by finding the k-Nearest Neighbors (kNN) of each residue, using the minimum pair-wise distance between all atoms in v_i and v_j:

$$d(v_i, v_j) = \min_{1 \le p \le c_i, 1 \le q \le c_j} ||\boldsymbol{X}_i(:, p) - \boldsymbol{X}_j(:, q)||_2, \quad (1)$$

where $\boldsymbol{X}_i(:, p)$ returns the p-th atom in $\boldsymbol{X}_i$ and $\boldsymbol{X}_i(:, q)$ is similarly defined. Inspired by Kong et al. (2022), we insert three global nodes into the heavy chain, the light chain, and the epitope, respectively, connecting to all nodes in their own chains. Besides, the global nodes of the heavy chain and the light chain are linked to each other.

Task Definition The residue vertices of the paratope are denoted as $\mathcal{V}_P$, clearly $\mathcal{V}_P \subseteq \mathcal{V}_A$. Given an epitope $\mathcal{G}_E(\mathcal{V}_E, \mathcal{E}_E)$ and an incomplete antibody sequence $\{s_i | i \in \mathcal{V}_A, i \notin \mathcal{V}_P\}$, we aim to design a model that simultaneously generates the 1D sequence of the paratope as well as the entire 3D structure of the antibody $(\mathcal{V}_A, \mathcal{E}_A)$ binding to the epitope, namely $\{s_i | i \in \mathcal{V}_P\}$ and $\{\boldsymbol{X}_i | v_i \in \mathcal{V}_A\}$.

4. Our Method: dyMEAN

The overall workflow of our dyMEAN is presented in Figure 3. During the calculation in dyMEAN, each vertex in epitope graph $\mathcal{G}_E$, antibody graph $\mathcal{G}_A$, and paratope subgraph $\mathcal{G}_P$ ($\mathcal{G}_P \subseteq \mathcal{G}_A$) is associated with an invariant vector $\boldsymbol{h}_i \in \mathbb{R}^d$ and an equivariant coordinate matrix $\boldsymbol{X}_i \in \mathbb{R}^{3 \times c_i}$. In form, the **overview of dyMEAN** is given by:

$$\mathcal{G}_A = \text{SI}(\{s_i\}_{i \in \mathcal{V}_A, i \notin \mathcal{V}_P}), i \in \mathcal{V}_A, \quad (2)$$

$$\mathcal{G}_S = \text{SP}(\mathcal{G}_E, \mathcal{G}_P) \quad (3)$$

$$\boldsymbol{h}_i, \boldsymbol{X}_i = \text{AME}(\mathcal{G}_E, \mathcal{G}_S, \mathcal{G}_A), i \in \mathcal{V}_E \cup \mathcal{V}_S \cup \mathcal{V}_A, \quad (4)$$

$$\boldsymbol{p}_i = \text{Predict}(\boldsymbol{h}_i), i \in \mathcal{V}_P, \quad (5)$$

$$\tilde{\boldsymbol{X}}_i = \text{Dock}(\mathcal{G}_A, \mathcal{G}_S), i \in \mathcal{V}_A, \quad (6)$$

where, SI (*a.k.a.* structural initialization) first initializes the coordinates $\boldsymbol{X}_i^{(0)}$ and the hidden states $\boldsymbol{h}_i^{(0)}$ for the antibody graph $\mathcal{G}_A$, based on the the incomplete antibody sequence; with the initialized paratope $\mathcal{G}_P$, SP (*a.k.a.* shadow paratope) attaches a shadow paratope $\mathcal{G}_S$, which shares the hidden states with the native one, to the epitope $\mathcal{G}_E$, creating a joint graph $\mathcal{G}_E \cup \mathcal{G}_S$. SP is crucial in bridging the epitope and the antibody for docking. Then, AME (*a.k.a.* adaptive multi-channel encoder) iteratively updates $\boldsymbol{X}_i$ and $\boldsymbol{h}_i$ for all vertices by message passing; finally, Eq. 5 predicts the distribution of amino acid types $\boldsymbol{p}_i$ for each paratope residue, and Eq. 6 docks the antibody $\mathcal{G}_A$ towards the shadow paratope $\mathcal{G}_S$, leading to the binding complex structure $\tilde{\boldsymbol{X}}_i$.

We will introduce the functions of SI, SP and AME in § 4.1, § 4.2 and § 4.3, respectively. The details of Eq. 5, Eq. 6, and training losses are provided in § 4.4. An elegant property of

dyMEAN is that its predicted paratope sequence is invariant and the binding structure is equivariant, with respect to the E(3) transformations (rotations/reflections/translations), making it well generalizable to different poses of the target epitope. We will mathematically reveal this point in § 4.4.

4.1. Structural Initialization with Conserved Residues

The input antibody sequence $\{s_i\}_{i \in \mathcal{V}_A, i \notin \mathcal{V}_P}$ involves neither paratope information nor the 3D geometry. Given such fragmentary information, this subsection investigates on how to attain desirable initialization for both $h_i^{(0)}$ and $X_i^{(0)}$.

Initializing $h_i^{(0)}$ We derive the initial embedding of each node via its amino acid type s_i and position number r_i in an numbering system ,*e.g.*, IMGT (Lefranc et al., 2003): $h_i^{(0)} = f(s_i, r_i) = f_{s_i} + f_{r_i}$, where f_{s_i} and f_{r_i} define the learnable amino acid embedding and position embedding, respectively. For the unknown paratope residue, we represent s_i by a special type [MASK].

Initializing $X_i^{(0)}$ We have the domain knowledge that the FRs of the antibody are well conserved (Klein et al., 2013) in spatial variation. It inspires us to first detect the well-conserved residues in FRs and then apply them to sketch the positions of other residues. While it is challenging to directly locate the conserved residues by comparing the residue-wise coordinates, we resort to the comparison of 1D sequences, which innately reflects the 3D spatial similarity (Jumper et al., 2021). To do so, we first align the antibody sequences in the dataset via a certain antibody numbering system (*e.g.*, IMGT). Then we consider a residue as well-conserved if its type is consistent among above 95% of the antibodies the analysis of different thresholds is provided in Appendix I). Next, we align all the antibodies by the backbone (i.e. N, C_α, C, O) coordinates of these well-conserved residues via Kabsch algorithm (Kabsch, 1976), and calculate the average backbone coordinates of these residues, leading to the backbone template $\{Z_{r_i} \in \mathbb{R}^{3 \times 4} | r_i \in \mathbb{W}\}$, where $\mathbb{W}$ collects the position numbers of the detected well-conserved residues. In our experiments, we identify 16 such residues in the heavy chain and 18 in the light chain. The backbone coordinates Z_i of other residues in the same chain are valued in this way: (1) for the ones between two nearest conserved residues in position number, we linearly interpolate their positions with unified spacing; (2) for those located at both ends of the chain, we conduct outwards linear-interpolation from the nearest conserved residue with the same interval used in the nearest pair of the residues computed in (1). More details are provided in Appendix A. Z_i is then extended to $X_i^{(0)}$ by filling α-carbon's coordinate in the side chains . We emphasize the significance of this knowledgeable initialization, which provides vague but essential guess of the antibody structure.

The coordinates are further normalized to conform to the standard Gaussian distribution $\mathcal{N}(0, I)$, by conducting 3D mean translation and 1D variance normalization (all dimensions of all antibodies share the same normalization factor to ensure consistent scale). After obtaining $X_i^{(0)}$, we construct the kNN edges for $\mathcal{G}_A$ via the distance defined in Eq.1.

4.2. E(3)-Invariant Attachment of Shadow Paratope

We attach a clone of the paratope around the epitope, which is called *shadow paratope*. It serves two crucial purposes in our end-to-end framework: (1) Transmitting E(3)-invariant information between the epitope and the antibody, by sharing the hidden states h_i and the same topology with the native paratope; (2) acting as the key points that will be used for the docking between the antibody and epitope, which will be detailed in § 4.4. One promising property of our shadow paratope attachment is that its 3D coordinates and final docked structure are independent of the initial position of the antibody, since it only exchanges the invariant information (*i.e.*, h_i not X_i) with the native paratope.

The shadow paratope subgraph is $\mathcal{G}_S = (\mathcal{V}_S, \mathcal{E}_S)$. Here, $\mathcal{E}_S$ contains two parts: internal edges copied from the connections between residues in the native paratope, and external edges linked to the epitope. For $v_i \in \mathcal{V}_E, v_j \in \mathcal{V}_S$, the external edges are constructed based on the kNN distance:

$$\hat{d}(v_i, v_j) = \phi_e(h_i, h_j) + \phi_e(h_j, h_i), \qquad (7)$$

where ϕ_e is a Multi-Layer Perceptron (MLP). The hidden vector h_i of $\mathcal{G}_S$ is duplicated from the native paratope, and the coordinates X_i are initialized around the center of the epitope according to standard Gaussian $\mathcal{N}(0, I)$[1]. $\mathcal{G}_S$ is merged into the epitope graph $\mathcal{G}_E$, creating $\mathcal{G}_E \cup \mathcal{G}_S$.

4.3. Adaptive Multi-Channel Equivariant Encoder

AME is able to handle X_i of different channel size, in order to consider full-atom geometry by involving side chains besides backbone atoms. This is why we call AME adaptive. The l-th layer updates the hidden vector h_i and the coordinate matrix X_i as follows:

$$m_{ij} = \phi_m(h_i^{(l)}, h_j^{(l)}, \frac{T_R(X_i^{(l)}, X_j^{(l)})}{||T_R(X_i^{(l)}, X_j^{(l)})||_F + \epsilon}), \quad (8)$$

$$X_{ij} = T_S(X_i^{(l)} - \frac{1}{c_j} \sum_{k=1}^{c_j} X_j^{(l)}(:, k), \phi_x(m_{ij})), \quad (9)$$

$$h_i^{(l+1)} = \phi_h(h_i^{(l)}, \sum_{j \in \mathcal{N}(i)} m_{ij}), \qquad (10)$$

$$X_i^{(l+1)} = X_i^{(l)} + \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} X_{ij} \qquad (11)$$

---

[1]The coordinates of the epitope have been normalized to $\mathcal{N}(0, I)$ beforehand in a similar way to the antibody in § 4.1.

where, ϕ_m, ϕ_x, ϕ_h are MLPs, $\mathcal{N}(i)$ denotes i's neighbors, $\boldsymbol{m}_{ij}$ and $\boldsymbol{X}_{ij}$ are non-geometric and geometric messages, respectively; the geometric relation extractor T_R and geometric message scaler T_S are for processing message between two distinct-shape matrices $\boldsymbol{X}_i \in \mathbb{R}^{3 \times c_i}$ and $\boldsymbol{X}_j \in \mathbb{R}^{3 \times c_j}$; the output of T_R is normalized with Frobenius norm following Huang et al. (2022), plus a constant $\epsilon = 1$ for numerical stability. Below are the details of T_R and T_S, and how information is exchanged between the epitope and the antibody.

Geometric Relation Extractor T_R Given $\boldsymbol{X}_i \in \mathbb{R}^{3 \times c_i}$ and $\boldsymbol{X}_j \in \mathbb{R}^{3 \times c_j}$, we first compute the channel-wise distance between each pair of the channels in $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$: $\boldsymbol{D}_{ij}(p, q) = ||\boldsymbol{X}_i(:, p) - \boldsymbol{X}_j(:, q)||_2$. Then, we employ two learnable weights $\boldsymbol{w}_i \in \mathbb{R}^{c_i \times 1}$ and $\boldsymbol{w}_j \in \mathbb{R}^{c_j \times 1}$ to characterize the channel-wise correlation in $\boldsymbol{D}_{ij}$, and two learnable attribute matrices $\boldsymbol{A}_i \in \mathbb{R}^{c_i \times d}$ and $\boldsymbol{A}_j \in \mathbb{R}^{c_j \times d}$ to extract useful patterns across each channel and output dimension (further details in Appendix B). The final output $\boldsymbol{R}_{ij} \in \mathbb{R}^{d \times d}$ is given by:

$$\boldsymbol{R}_{ij} = \boldsymbol{A}_i^\top (\boldsymbol{w}_i \boldsymbol{w}_j^\top \odot \boldsymbol{D}_{ij}) \boldsymbol{A}_j. \quad (12)$$

Clearly, $\boldsymbol{R}_{ij}$ keeps the same shape regardless of the change in c_i or c_j, namely static-dimensional inputs for ϕ_m and ϕ_h.

Geometric Message Scaler T_S The main purpose of T_S is to generate geometric messages by scaling the input coordinates $\boldsymbol{X} \in \mathbb{R}^{3 \times c}$ with the non-geometric message $\boldsymbol{s} = \phi_x(\boldsymbol{m}_{ij}) \in \mathbb{R}^C$ where C is the upper bound of the channel size. In detail, $T_S(\boldsymbol{X}, \boldsymbol{s})$ is calculated by:

$$\boldsymbol{X}' = \boldsymbol{X} \cdot \operatorname{diag}(\boldsymbol{s}'), \quad (13)$$

where $\boldsymbol{s}' \in \mathbb{R}^c$ is the average pooling of $\boldsymbol{s}$ with the window size $C - c + 1$ and stride 1, $\operatorname{diag}(\cdot)$ returns the matrix with the input vector as the diagonal elements, and thus the output $\boldsymbol{X}'$ shares the same shape with $\boldsymbol{X}$.

Information Exchanging between $\mathcal{G}_E$ and $\mathcal{G}_A$ Although the epitope graph $\mathcal{G}_E$ and the antibody graph $\mathcal{G}_A$ are disconnected, their information is exchanged via the hidden states of the shadow paratope $\mathcal{G}_S$. In particular, we first conduct 1-layer AME on $\mathcal{G}_A$ and copy the hidden vectors $\boldsymbol{h}_i$ from the native paratope $\mathcal{G}_P$ to the shadow paratope $\mathcal{G}_S$. Then, we carry out 1-layer AME on $\mathcal{G}_E \cup \mathcal{G}_S$ and copy the hidden vectors reversely from $\mathcal{G}_S$ to $\mathcal{G}_P$. The above two stages are alternated until L layers. We additionally run 1-layer message passing on $\mathcal{G}_A$ to broadcast the updated information across the entire antibody.

Nicely, T_R is E(3)-invariant, T_S is O(3)-equivariant, and the information exchanging between $\mathcal{G}_E$ and $\mathcal{G}_A$ is E(3)-invariant, therefore for the final outputs of AME, $\boldsymbol{h}_i$ is E(3)-invariant and $\boldsymbol{X}_i$ is independently E(3)-equivariant (Ganea et al., 2021) w.r.t. $\mathcal{G}_E \cup \mathcal{G}_S$ and $\mathcal{G}_A$. Such property will permit E(3)-invariance of dyMEAN stated in Theorem 4.1.

4.4. Prediction, Docking and Training Losses

Prediction   With the output by AME, we leverage the progressive full-shot decoding strategy from Kong et al. (2022) to generate the 1D sequence and the 3D structure over T iterations. To be specific, each iteration updates the hidden states and the coordinates for all vertices:

$$\{\boldsymbol{h}_i^{(t)}, \boldsymbol{X}_i^{(t)}\} = \operatorname{AME}(\{\boldsymbol{h}_i^{(t-1)}, \boldsymbol{X}_i^{(t-1)}\}). \quad (14)$$

We predict the amino acid type of the paratope with $\boldsymbol{h}_i^{(t)}$, :

$$\boldsymbol{p}_i^{(t)} = \operatorname{Softmax}(\phi_p(\boldsymbol{h}_i^{(t)})), i \in \mathcal{V}_P, \quad (15)$$

where ϕ_p is an MLP. The hidden states are refreshed as:

$$\boldsymbol{h}_i^{(t)} = \begin{cases} \boldsymbol{f}(s_i, r_i) + \phi_d(\boldsymbol{h}_i^t), & i \notin \mathcal{V}_P, \\ \sum_{j=1}^{n_a} p_{i,j}^{(t)} \boldsymbol{f}(s_j, r_i) + \phi_d(\boldsymbol{h}_i^{(t)}), & i \in \mathcal{V}_P, \end{cases} \quad (16)$$

where the embedding $\boldsymbol{f}(s_i, r_i) = \boldsymbol{f}_{s_i} + \boldsymbol{f}_{r_i}$, ϕ_d is an MLP, n_a is the number of amino acid types, and $p_{i,j}^{(t)}$ returns the j-th element of $\boldsymbol{p}_i^{(t)}$. The second line aims at performing soft smoothing of the embeddings with the predicted probability $\boldsymbol{p}_i^{(t)}$. Compared with MEAN (Kong et al., 2022), the memory term $\phi_d(\boldsymbol{h}_i^{(t)})$ is extra added for better information reservation, which will be ablated in our experiments.

The new $\boldsymbol{h}_i^{(t)}$, along with $\boldsymbol{X}_i^{(t)}$ from Eq. 14 will be used as the input for the next iteration. After each iteration, we recreate the edges $\mathcal{E}_E, \mathcal{E}_A, \mathcal{E}_S$ by calculating the distance in Eq. 1 and Eq. 7 based on the current values $\boldsymbol{X}_i^{(t)}$ and $\boldsymbol{h}_i^{(t)}$.

Docking   After the final iteration, we align the pose of the native paratope with the shadow paratope via Kabsch algorithm (Kabsch, 1976). The docked coordinates $\{\tilde{\boldsymbol{X}}_i | v_i \in \mathcal{V}_A\}$ are given by:

$$\boldsymbol{Q}, \boldsymbol{t} = \operatorname{Kabsch}(\{\boldsymbol{X}_i^{(T)} | i \in \mathcal{V}_P\}, \{\boldsymbol{X}_i^{(T)} | i \in \mathcal{V}_S\}), \quad (17)$$

$$\tilde{\boldsymbol{X}}_i = \boldsymbol{Q} \boldsymbol{X}_i^{(T)} + \boldsymbol{t}, v_i \in \mathcal{V}_A, \quad (18)$$

where $\boldsymbol{Q} \in \mathrm{O}(3), \boldsymbol{t} \in \mathbb{R}^3$, and O(3) is the orthogonal group.

Loss Function   The loss function sums up the three parts: sequence loss $\mathcal{L}_{\text{seq}}$, structure loss $\mathcal{L}_{\text{struct}}$ and docking loss $\mathcal{L}_{\text{dock}}$. The cross-entropy loss ℓ_{ce} is utilized to guide the sequence prediction at each iteration:

$$\mathcal{L}_{\text{seq}} = \frac{1}{T|\mathcal{V}_P|} \sum_{t=1}^T \sum_{i \in \mathcal{V}_P} \ell_{\text{ce}}(\boldsymbol{p}_i^{(t)}, \boldsymbol{p}_i^\star). \quad (19)$$

For structure supervision, we exert Huber loss (Huber, 1992) on the coordinates of the final iteration. As suggested by Kong et al. (2022), Huber loss maintains numerical stability for noisy data (further details in Appendix E):

$$\mathcal{L}_{\text{coord}} = \frac{1}{|\mathcal{V}_A|} \sum_{v_i \in \mathcal{V}_A} \ell_{\text{huber}}(\boldsymbol{X}_i^{(T)}, \boldsymbol{X}_i^\star), \quad (20)$$

where $X_i^\star$ denotes the ground-truth coordinates aligned to $X_i^{(T)}$ by Kabsch algorithm. Since our method generates the structure of all atoms, we further supervise bond lengths to capture the local geometry:

$$\mathcal{L}_{\text{bond}} = \frac{1}{|\mathcal{B}|} \sum_{b \in \mathcal{B}} \ell_{\text{huber}}(b^{(T)}, b^\star), \tag{21}$$

where $\mathcal{B}$ contains all chemical bonds in the antibody, $b^{(T)}$ and $b^\star$ denote the bond length derived from $X_i^{(T)}$ and the ground truth, respectively. The structure loss is the sum of the above two losses: $\mathcal{L}_{\text{struct}} = \mathcal{L}_{\text{coord}} + \mathcal{L}_{\text{bond}}$.

For docking, it is sufficient to supervise the shadow paratope by the coordinate loss and the external distance loss:

$$\mathcal{L}_{\text{sp}} = \frac{1}{|\mathcal{V}_S|} \sum_{i \in \mathcal{V}_S} \ell_{\text{huber}}(X_i^{(T)}, X_i^\star), \tag{22}$$

$$\mathcal{L}_{\text{dist}} = \frac{1}{T|\mathcal{V}_E||\mathcal{V}_S|} \sum_{t=1}^{T} \sum_{u \in \mathcal{V}_E, v \in \mathcal{V}_S} \ell_{\text{huber}}(\hat{d}^{(t)}(u, v), d^\star(u, v)), \tag{23}$$

where $\hat{d}^{(t)}$ defined in Eq. 7 computes the external edge distance at t-th iteration and $d^\star$ is the ground-truth distance. The docking loss becomes: $\mathcal{L}_{\text{dock}} = \mathcal{L}_{\text{sp}} + \mathcal{L}_{\text{dist}}$.

We now demonstrate an elegant property of our dyMEAN: it is E(3)-equivariant with respect to the initial position and orientation of the epitope.

Theorem 4.1. *Given the initial epitope (along with the shadow paratope) $\{h_i, X_i\}_{i \in \mathcal{V}_E \cup \mathcal{V}_S}$ and the initialized antibody $\{h_i^{(0)}, X_i^{(0)}\}_{i \in \mathcal{V}_A}$, we compute the final prediction and docking by $\{p_i\}_{i \in \mathcal{V}_P}, \{\tilde{X}_i\}_{i \in \mathcal{V}_A} = dyMEAN\left(\{h_i, X_i\}_{i \in \mathcal{V}_E \cup \mathcal{V}_S}, \{h_i^{(0)}, X_i^{(0)}\}_{i \in \mathcal{V}_A}\right)$. We immediately have the conclusion that dyMEAN is E(3)-equivariant. Namely, for any transformations $g_1, g_2 \in E(3)$, we have $\{p_i\}_{i \in \mathcal{V}_P}, \{g_1 \cdot \tilde{X}_i\}_{i \in \mathcal{V}_A} = dyMEAN\left(\{h_i, g_1 \cdot X_i\}_{i \in \mathcal{V}_E \cup \mathcal{V}_S}, \{h_i^{(0)}, g_2 \cdot X_i^{(0)}\}_{i \in \mathcal{V}_A}\right)$ where $g \cdot X := QX + t$ for orthogonal transformation $Q \in O(3)$ and translation transformation $t \in \mathbb{R}^3$.*

The proof is provided in Appendix C. This theorem is crucial, as it tells that our dyMEAN is well generalizable to arbitrary orientation and position of the epitope as well as the initialized antibody, and it is thus data-efficient.

5. Experiments

We conduct experiments on the three tasks: (1) Epitope-binding CDR-H3 generation (§ 5.1); (2) Complex structure prediction (§ 5.2); (3) Affinity optimization (§ 5.3). We also try designing binders on general proteins and provide the results in Appendix L. Following Kong et al. (2022), we extract the 48 residues closest to the antibody as the epitope, which is sufficient to include all binding residues in the antigen (Shan et al., 2022).

Since there is no previous method for end-to-end full-atom antibody design, we implement each subtask of the whole pipeline (structure prediction⇒docking⇒CDR generation⇒side-chain packing) with existing competitive approaches. For antibody structure prediction, we select the official implement of **IgFold** (Ruffolo & Gray, 2022) that is a specialization of AlphaFold (Evans et al., 2022) for the antibody domain. For docking, we leverage **HDock** (Yan et al., 2020), which is a prevailing model with knowledge-based scoring functions. For CDR generation, the following baselines are implemented: **RosettaAb** (Adolf-Bryfogle et al., 2018) searches for the optimal sequence and structure guided by statistical energy functions; **MEAN** (Kong et al., 2022) generates both 1D sequences and 3D structures with equivariant attention graph networks; **Diffab** (Luo et al., 2022) is a diffusion-based generative model and has considered side-chain orientations. To further involve side chains, we use Rosetta (Alford et al., 2017) to cope with side-chain packing, which is also a built-in step of RosettaAb. Besides, we implement **HERN** (Jin et al., 2022) that needs no external structure prediction, docking, and side-chain packing but is unaware of framework region modeling and inefficient in autoregressive generation of all atoms. Further implementation details are deferred to Appendix F.

5.1. Epitope-binding CDR-H3 Generation

The experiments here test the central goal of end-to-end antibody design, as illustrated in Figure 1. As CDR-H3 is the most variant region among all CDRs and largely determines the binding specificity and affinity (Raybould et al., 2019), we consider it as the paratope to be generated. We also provide the analysis for designing multiple CDRs as well as the entire antibody in § 6.

We use the following metrics for quantitative assessment: **Amino Acid Recovery (AAR)** is defined as the overlapping ratio of the generated sequence and the ground truth; **CAAR** (Ramaraj et al., 2012) computes AAR restricted to binding residues whose minimum distance from epitope residues is below 6.6 Å; **TMscore** (Zhang & Skolnick, 2004; Xu & Zhang, 2010) measures the global similarity between the generated structure and the ground truth in terms of C_α coordinates; **Local Distance Difference Test (lDDT)** (Mariani et al., 2013) contrasts the difference of the atom-wise distance matrix between the generated structure and the ground truth; **RMSD** calculates the Root Mean Square Deviation regarding the absolute coordinates of CDR-H3 without Kabsch alignment; **DockQ** (Basu & Wallner, 2016) is a comprehensive score for the docking quality. Both TM-score and lDDT range from 0 to 1 and are invariant to E(3)-transformations of the antibody structure, while RMSD and

Table 1. Results of epitope-binding CDR-H3 design on RAbD. Methods with superscript ∗ adopt the pipeline: IgFold ⇒ HDock ⇒ CDR generation ⇒ Rosetta side-chain packing.

Model	Generation			Docking		
	AAR↑	TMscore↑	lDDT↑	CAAR↑	RMSD↓	DockQ↑
RosettaAb*	32.31%	0.9717	0.8272	14.58%	17.70	0.137
DiffAb*	35.31%	0.9695	0.8281	22.17%	23.24	0.158
MEAN*	37.38%	0.9688	0.8252	24.11%	17.30	0.162
HERN	32.65%	-	-	19.27%	9.15	0.294
Initialization	-	0.5072	0.2998	-	-	-
dyMEAN	**43.65%**	**0.9726**	**0.8454**	**28.11%**	**8.11**	**0.409**

DockQ focus on the docking quality and are sensitive to the relative position of the antibody to the epitope.

We train all models on the Structural Antibody Database (SAbDab, Dunbar et al., 2014) retrieved in November 2022, and assess them with the RAbD benchmark (Adolf-Bryfogle et al., 2018) composed of 60 diverse complexes selected by domain experts. We split SAbDab into the training and validation sets with a ratio of 9 : 1 according to CDR-H3 clusters as suggested by Jin et al. (2021); Kong et al. (2022). Each cluster is formed by antibodies sharing above 40% CDR-H3 sequence identity calculated by the BLOSUM62 substitution matrix (Henikoff & Henikoff, 1992). The antibodies in the same clusters as the test set are dropped to maintain a convincing generalization test. We implement the clustering process with MMseqs2 (Steinegger & Söding, 2017) and the numbers of antibodies (clusters) in the training and the validation sets are 3,256 (1,644) and 365 (182).

Results As shown in Table 1, our dyMEAN remarkably outperforms all baselines regarding nearly all metrics, supporting its superiority in recovering 1D sequences, 3D structures, and the binding interface. In contrast to the pipeline-based models (RosettaAb*, DiffAb*, and MEAN*), dyMEAN is end-to-end and able to alleviate potential accumulated errors incurred by each stage of the antibody design process, hence leading to better performance. Compared with HERN which is unaware of frame region modeling, dyMEAN is clearly more advantageous in both 1D generation and docking, indicating that characterizing the full-context geometry in antibody design is useful and even indispensable. In addition, the TMscore and lDDT of the initialized structure via SI are meaningful but still far from satisfactory, which explains the importance of later message passing by AME in dyMEAN. As an illustrated example, Figure 4 visualizes the comparison between MEAN* and dyMEAN. More samples are provided in Appendix M. We further analyze the distribution of the χ-angles of the generated side chains in Appendix H.

5.2. Complex Structure Prediction

This task predicts the docked complex structure given the complete antibody sequence (including CDR-H3). We report the metrics of TMscore, lDDT, RMSD, and DockQ.

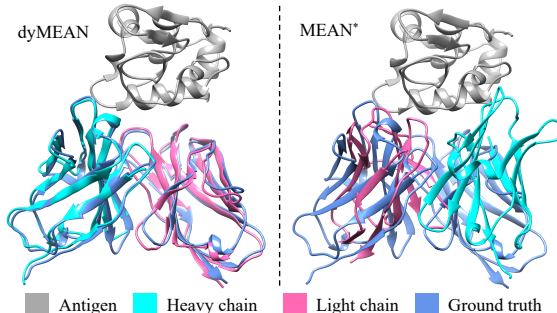| Antigen | Heavy chain | Light chain | Ground truth |

Figure 4. Complexes (pdb: 1ic7) generated by our dyMEAN (DockQ= 0.971) and MEAN* (DockQ= 0.046).

As there is no need for CDR generation, the pipeline-based method is reduced as: IgFold⇒Hdock⇒ Rosetta. To better depict the effectiveness of our method, we also implement the docking version of HERN in two considerate ways: (1) taking input as the backbone structure predicted by IgFold, HERN outputs the docked backbones, followed by Rosetta for side-chain packing; (2) taking input as the ground-truth antibody structures, HERN docks CDR-H3 along with other regions towards the epitope. We train all models on SAbDab with the training-validation ratio of 9:1 and evaluate on the test set (51 antigen-antibody complexes) used in IgFold paper (Ruffolo & Gray, 2022) to avoid any potential data leakage when applying IgFold during testing.

Results Table 2 reads that dyMEAN surpasses all other methods in terms of both structure prediction and docking. Excitingly, though IgFold leverages embeddings from a pretrained antibody language model (Ruffolo et al., 2021) and utilizes 38k additional antibody structures from AlphaFold (Jumper et al., 2021) for training, our model still achieves better TMscore and lDDT, exhibiting its stronger capability of learning the distribution of antibody structures. As for the baseline GT⇒HERN that applies ground-truth structures for docking, our dyMEAN still yields better docking accuracy, which reveals that dyMEAN really excels at unveiling the epitope-antibody interactions with the full-context geometry. We also explore including other CDRs into the shadow paratope which is presented in Appendix J.d

Table 2. Complex Structure Prediction. Methods with superscript ∗ use Rosetta to generate the side chains. The values with † are the upper-bound as they are calculated on Ground Truths (GT).

Model	Structure		Docking	
	TMscore↑	lDDT↑	RMSD↓	DockQ↑
IgFold⇒HDock*	0.9701	0.8439	16.32	0.202
IgFold⇒HERN*	0.9702	0.8441	9.63	0.429
GT⇒HERN	1.0000†	1.0000†	9.65	0.432
initialization	0.5054	0.3006	-	-
dyMEAN	**0.9731**	**0.8673**	**9.05**	**0.452**

5.3. Affinity Optimization

Another common application is to optimize the affinity of a given antibody. As suggested by Kong et al. (2022), we use

Table 3. Average $\Delta\Delta G$ (kcal/mol) and average number of changed residues (ΔL). dyMEAN-n denotes the restricted version allowing at most n changed residues, and dyMEAN itself changes n residues with n is sampled from $[1, N]$ at each generation.

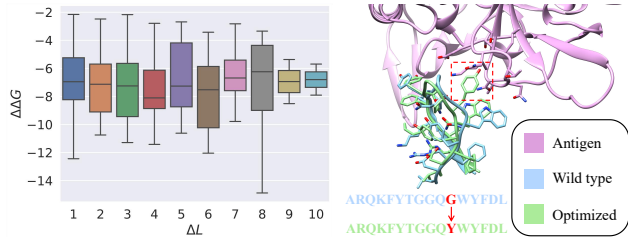| Method	$\Delta\Delta G \downarrow$	$\Delta L \downarrow$	Method	$\Delta\Delta G \downarrow$	$\Delta L \downarrow$
Diffab ($t=1$)	-0.32	1.19	MEAN	-6.48	8.96
Diffab ($t=2$)	-0.68	1.21	dyMEAN-1	-6.79	1.00
Diffab ($t=4$)	-1.00	1.38	dyMEAN-2	-7.11	1.59
Diffab ($t=8$)	-1.34	1.62	dyMEAN-4	-7.18	3.24
Diffab ($t=16$)	-1.85	3.54	dyMEAN-8	-7.23	6.67
Diffab ($t=32$)	-2.17	7.06	dyMEAN	**-7.31**	5.57

Figure 5. Left: The distribution of $\Delta\Delta G$ w.r.t. the actual ΔL of the candidates after optimization. Right: The binding interface of an optimized antibody (pdb: 3se9, $\Delta\Delta G = -7.22$) with only one residue changed compared to the wild type.

the binding affinity change ($\Delta\Delta G$) as the objective, which is predicted by a GNN-based predictor (Shan et al., 2022). We also provide the results using FoldX (Schymkowitz et al., 2005) as the affinity predictor in Appendix K. We conduct evaluation on the antibodies from SKEMPI V2.0 (Jankauskaitė et al., 2019). We also report the number of changed residues ΔL since many practical scenarios prefer smaller ΔL (Ren et al., 2022). To adjust dyMEAN for this task, we additionally train an MLP over the representations of the complex graphs to fit the above-mentioned $\Delta\Delta G$ predictor. Then we conduct gradient search to locate favorable initial states of all residues, which are likely to generate a complex of higher affinity. A few more adaptions are needed, which are detailed in Appendix D. For compared baselines, we use ITA for MEAN, and the intermediate state at the $(T-t)$-th step during the denoising process for DiffAb, as suggested in their papers. All models are trained on SAbDab under the same settings as § 5.1. For each antibody in the test set, we generate 100 candidates and record the $\Delta\Delta G$ of the top-1 candidate, and then compute the corresponding ΔL.

Results Table 3 summarizes the average $\Delta\Delta G$ and ΔL over all test antibodies. It shows that dyMEAN generates antibodies with the lowest $\Delta\Delta G$ and controllable changes of ΔL. Although DiffAb can also control ΔL by reducing t, its ability to affinity optimization is limited. MEAN achieves favorable $\Delta\Delta G$ but at the cost of great change in ΔL. It is worth mentioning that our model still achieves desirable performance even when only 1 or 2 residues are allowed to change. Figure 5 (right) illustrates one example in this case.

Table 4. Ablations of different components in dyMEAN.

Model	Generation			Docking		
	AAR↑	TMscore↑	LDDT↑	CAAR↑	RMSD↓	DockQ↑
CDR-H3 Design						
dyMEAN	**43.65%**	0.9726	**0.8454**	28.11%	**8.11**	**0.409**
$T=2$	43.57%	**0.9731**	0.8411	**29.23%**	9.68	0.383
$T=4$	42.84%	0.9725	0.8440	28.18%	8.65	0.393
- full-atom	41.81%	0.9730	0.7999	27.96%	10.10	0.343
- sharing	43.17%	0.9718	0.8374	28.79%	9.46	0.356
- $\boldsymbol{w}_i\boldsymbol{w}_j^\top$	39.29%	0.9724	0.8408	25.87%	8.60	0.407
- memory	40.01%	0.9727	0.8444	24.37%	9.03	0.378
- $\mathcal{L}_{\text{dist}}$	42.32%	0.9715	0.8361	27.46%	9.07	0.393
Complex Structure Prediction						
dyMEAN	-	**0.9731**	**0.8673**	-	9.05	**0.452**
$T=2$	-	0.9716	0.8606	-	9.61	0.440
$T=4$	-	0.9712	0.8628	-	9.62	0.441
- full-atom	-	0.9713	0.8111	-	9.98	0.424
- sharing	-	0.9709	0.8641	-	11.27	0.429
- $\boldsymbol{w}_i\boldsymbol{w}_j^\top$	-	0.9725	0.8662	-	9.16	0.432
- memory	-	0.9711	0.8653	-	**8.89**	0.447
- $\mathcal{L}_{\text{dist}}$	-	0.9706	0.8587	-	9.97	0.416

6. Analysis

Ablation Study We ablate the necessity or value of the following components: the number of iterations in generation T, the full-atom geometry, the information sharing between the shadow and the native paratope, the learnable channel weights $\boldsymbol{w}_i\boldsymbol{w}_j^\top$ in Eq. 12, the memory term ϕ_d in Eq. 16, and the external distance prediction loss $\mathcal{L}_{\text{dist}}$ in Eq. 23. Particularly, for the ablation of the full-atom geometry, we only retain backbone atoms in dyMEAN and use Rosetta for side-chain packing afterward; for $\mathcal{L}_{\text{dist}}$, we instead use the coordinates to compute distances between residue pairs other than predicting them with hidden states. Table 4 presents the following observations: (1) The value of T mainly affects the docking performance, and $T=3$ used in dyMEAN generally yields the best performance. (2) The removal of the full-atom geometry exerts a remarkably adverse impact on the overall performance, which confirms the necessity of incorporating the side-chain conformation. (3) The information sharing is critical specifically for structure generation and docking, without which all metrics excluding CAAR drop by a large margin. (4) The learnable weights act like attentions to different channels, and will incur detriment if removed. (5) The memory mechanism contributes to the task of CDR-H3 design but seems nonessential on complex structure prediction, which is reasonable because the information passed for the hidden states of CDR-H3 sequence in Eq. 16 is closely influenced by this term while the 3D coordinates are directly passed on to the next iteration. (6) The external distance prediction loss $\mathcal{L}_{\text{dist}}$ is vital for docking, which we suspect the coordinates alone are insufficient to correctly recover the structure of the shadow paratope specifically during early iterations.

Multiple CDRs Design and Full Antibody Design In § 5.1, we follow previous settings (e.g. HERN) and focuses mainly on the design of CDR-H3, because CDR-H3 is the loop mostly involved in binding and the most difficult to model. However, our method can be easily extended to

include other CDRs or any target regions. What we need to do is just masking all of them to generate, while the overall generation architecture keeps the same. To illustrate this flexibility, we additionally extend our model to the simultaneous design of all 6 CDRs and report the results in Table 5. It suggests that the results are promising in general.

Further, we provide the results for designing the full antibody including the framework regions in Table 5. It reads that AAR improves by a large margin, which is expected because the other parts excluding CDR-H3 in an antibody exhibit stronger regularities and conservativeness. The side-chain generation (lDDT) worsens, which is also expected because it is more challenging to simultaneously generate the type of the residue as well as its side-chain geometry than generating the side chains with known residue type in the framework regions. The performance on backbone generation (TMscore) and docking (DockQ) remain similar to the CDR-H3 design experiment (§ 5.1).

Table 5. Evaluation on designing all 6 CDRs simultaneously and designing the full antibody.

	AAR	TMscore	lDDT	DockQ
Simultaneous Design of 6 CDRs				
AAR details				
L1	75.55%		H1	75.72%
L2	83.10%		H2	68.48%
L3	52.12%		H3	37.51%
All	60.07%	0.9653	0.8029	0.396
Design of Full Antibody				
Full	74.96%	0.9662	0.7589	0.412

7. Limitations

Data Diversity and Evaluation Metrics Currently, deep generative models are likely to face difficulties in antibody design due to the limited diversity of antigen-antibody data. We count the most frequent unigram pattern of the amino acid types of each position in CDR-H3 from the training set, matching from both sides to the middle, which yields the pattern $ARDG * * * DY$ where most $*$ are Y. We use this unigram pattern to calculate AAR on the test set and obtain AAR= 39.61% and CAAR= 26.57%. This implies that the meaningless unigram pattern is prevailing in both the training set and the test set, which may hinder the models from learning meaningful antigen-antibody interaction patterns and trick the evaluation metrics. After removing the first 4 residues and the last 2 residues from CDR-H3, dyMEAN achieves an AAR of 31.76%, which exhibits a clear performance detriment compared to Table 1. These phenomena encourage future work in augmenting the dataset (*e.g.* through wet-lab experiments or extracting similar interfaces from general protein complexes) and proposing better evaluation metrics to avoid the impact of the unigram distribution (*e.g.* drop the residues that can be predicted correctly by unigram

pattern).

Reliability of Computational Energy Functions Ultimately the binding affinity (or binding energy) determines whether the generated candidates are good binders or not. In this paper, we use the deep-learning-based predictor of $\Delta\Delta G$, and it is also common to use statistical energy terms (*e.g.* FoldX (Schymkowitz et al., 2005), Rosetta (Alford et al., 2017), docking scores in softwares (Goodsell et al., 1996)). However, the reliability of current computational energy functions still remains uncertain, and some are known to correlate poorly with the experimental results (Ramírez & Caballero, 2016; 2018). Indeed, there are two questions required to answer: (1) If these energy functions which are fitted on well binding complexes can distinguish poorly binding complexes? (2) If these energy functions which are fitted on natural complexes can generalize to complexes generated by deep models which may yield distinct distributions? We believe a well-generalizable affinity predictor is essential for learning-based antibody design; otherwise, wet-lab evaluations are necessary, which, yet, are inefficient and labor-intensive.

8. Conclusion

In this paper, we propose dyMEAN, a full-atom model for end-to-end antibody design given the epitope and the incomplete antibody sequence. Specifically, we explore a knowledge-guided structural initialization and propose shadow paratope for E(3)-equivariant message passing and docking. The proposed adaptive multi-channel encoder also tackles the challenge of the variant number of atoms in different residues in full-atom modeling. Our dyMEAN surpasses state-of-the-art models in terms of epitope-binding CDR-H3 design, complex structure prediction, and affinity optimization. Our work provides insights into the end-to-end antibody design and could inspire future research on the full-atom modeling of proteins.

Reproducibility

Codes for our dyMEAN are available at https://github.com/THUNLP-MT/dyMEAN.
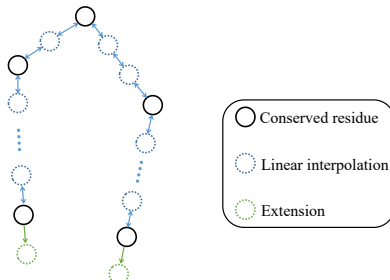
Acknowledgments

References

Adolf-Bryfogle, J., Kalyuzhniy, O., Kubitz, M., Weitzner, B. D., Hu, X., Adachi, Y., Schief, W. R., and Dunbrack Jr, R. L. Rosettaantibodydesign (rabd): A general framework for computational antibody design. *PLoS computational biology*, 14(4):e1006112, 2018.

Akbar, R., Robert, P. A., Pavlović, M., Jeliazkov, J. R., Snapkov, I., Slabodkin, A., Weber, C. R., Scheffer, L., Miho, E., Haff, I. H., et al. A compact vocabulary of paratope-epitope interactions enables predictability of antibody-antigen binding. *Cell Reports*, 34(11):108856, 2021.

Akbar, R., Robert, P. A., Weber, C. R., Widrich, M., Frank, R., Pavlović, M., Scheffer, L., Chernigovskaya, M., Snapkov, I., Slabodkin, A., et al. In silico proof of principle of machine learning-based antibody design at unconstrained scale. In *Mabs*, volume 14, pp. 2031482. Taylor & Francis, 2022.

Alford, R. F., Leaver-Fay, A., Jeliazkov, J. R., O'Meara, M. J., DiMaio, F. P., Park, H., Shapovalov, M. V., Renfrew, P. D., Mulligan, V. K., Kappel, K., et al. The rosetta all-atom energy function for macromolecular modeling and design. *Journal of chemical theory and computation*, 13 (6):3031–3048, 2017.

Almagro, J. C., Daniels-Wells, T. R., Perez-Tapia, S. M., and Penichet, M. L. Progress and challenges in the design and clinical development of antibodies for cancer therapy. *Frontiers in immunology*, 8:1751, 2018.

Basu, S. and Wallner, B. Dockq: a quality measure for protein-protein docking models. *PLoS one*, 11(8): e0161879, 2016.

Carter, P. J. Potent antibody therapeutics by design. *Nature reviews immunology*, 6(5):343–357, 2006.

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

Eastman, P., Swails, J., Chodera, J. D., McGibbon, R. T., Zhao, Y., Beauchamp, K. A., Wang, L.-P., Simmonett, A. C., Harrigan, M. P., Stern, C. D., et al. Openmm 7: Rapid development of high performance algorithms for molecular dynamics. *PLoS computational biology*, 13(7): e1005659, 2017.

Evans, R., O'Neill, M., Pritzel, A., Antropova, N., Senior, A., Green, T., Žídek, A., Bates, R., Blackwell, S., Yim, J., et al. Protein complex prediction with alphafold-multimer. *BioRxiv*, pp. 2021–10, 2022.

Foote, J. and Winter, G. Antibody framework residues affecting the conformation of the hypervariable loops. *Journal of molecular biology*, 224(2):487–499, 1992.

Fuchs, F., Worrall, D., Fischer, V., and Welling, M. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in Neural Information Processing Systems*, 33:1970–1981, 2020.

Ganea, O.-E., Huang, X., Bunne, C., Bian, Y., Barzilay, R., Jaakkola, T., and Krause, A. Independent se (3)-equivariant models for end-to-end rigid protein docking. *arXiv preprint arXiv:2111.07786*, 2021.

Gasteiger, J., Groß, J., and Günnemann, S. Directional message passing for molecular graphs. *arXiv preprint arXiv:2003.03123*, 2020.

Goodsell, D. S., Morris, G. M., and Olson, A. J. Automated docking of flexible ligands: applications of autodock. *Journal of molecular recognition*, 9(1):1–5, 1996.

Henikoff, S. and Henikoff, J. G. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22):10915–10919, 1992.

Huang, W., Han, J., Rong, Y., Xu, T., Sun, F., and Huang, J. Equivariant graph mechanics networks with constraints. *arXiv preprint arXiv:2203.06442*, 2022.

Huber, P. J. Robust estimation of a location parameter. In *Breakthroughs in statistics*, pp. 492–518. Springer, 1992.

IUPAC, I. et al. Abbreviations and symbols for the description of the conformation of polypeptide chains. *Biochemistry*, 9:3471–3479, 1970.

Jankauskaitė, J., Jiménez-García, B., Dapkūnas, J., Fernández-Recio, J., and Moal, I. H. Skempi 2.0: an updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation. *Bioinformatics*, 35(3):462–469, 2019.

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

Jin, W., Barzilay, R., and Jaakkola, T. Antibody-antigen docking and design via hierarchical structure refinement. In *International Conference on Machine Learning*, pp. 10217–10227. PMLR, 2022.

Jones, S. and Thornton, J. M. Principles of protein-protein interactions. *Proceedings of the National Academy of Sciences*, 93(1):13–20, 1996.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

Kabsch, W. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923, 1976.

Klein, F., Diskin, R., Scheid, J. F., Gaebler, C., Mouquet, H., Georgiev, I. S., Pancera, M., Zhou, T., Incesu, R.-B., Fu, B. Z., et al. Somatic mutations of the immunoglobulin framework are generally required for broad and potent hiv-1 neutralization. *Cell*, 153(1):126–138, 2013.

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

Kozakov, D., Hall, D. R., Xia, B., Porter, K. A., Padhorny, D., Yueh, C., Beglov, D., and Vajda, S. The cluspro web server for protein–protein docking. *Nature protocols*, 12 (2):255–278, 2017.

Kullback, S. and Leibler, R. A. On information and sufficiency. *The annals of mathematical statistics*, 22(1): 79–86, 1951.

Kuroda, D., Shirai, H., Jacobson, M. P., and Nakamura, H. Computer-aided antibody design. *Protein engineering, design & selection*, 25(10):507–522, 2012.

Leaver-Fay, A., Tyka, M., Lewis, S. M., Lange, O. F., Thompson, J., Jacak, R., Kaufman, K. W., Renfrew, P. D., Smith, C. A., Sheffler, W., et al. Rosetta3: an object-oriented software suite for the simulation and design of macromolecules. In *Methods in enzymology*, volume 487, pp. 545–574. Elsevier, 2011.

Lefranc, M.-P., Pommié, C., Ruiz, M., Giudicelli, V., Foulquier, E., Truong, L., Thouvenin-Contet, V., and Lefranc, G. Imgt unique numbering for immunoglobulin and t cell receptor variable domains and ig superfamily v-like domains. *Developmental & Comparative Immunology*, 27(1):55–77, 2003.

Li, T., Pantazes, R. J., and Maranas, C. D. Optmaven–a new framework for the de novo design of antibody variable region models targeting specific antigen epitopes. *PloS one*, 9(8):e105954, 2014.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637): 1123–1130, 2023.

Liu, G., Zeng, H., Mueller, J., Carter, B., Wang, Z., Schilz, J., Horny, G., Birnbaum, M. E., Ewert, S., and Gifford, D. K. Antibody complementarity determining region design using high-capacity machine learning. *Bioinformatics*, 36(7):2126–2133, 2020.

Luo, S., Su, Y., Peng, X., Wang, S., Peng, J., and Ma, J. Antigen-specific antibody design and optimization with diffusion-based generative models. *bioRxiv*, 2022.

MacCallum, R. M., Martin, A. C., and Thornton, J. M. Antibody-antigen interactions: contact analysis and binding site topography. *Journal of molecular biology*, 262 (5):732–745, 1996.

MacKerell Jr, A. D., Brooks, B., Brooks III, C. L., Nilsson, L., Roux, B., Won, Y., and Karplus, M. Charmm: the energy function and its parameterization. *Encyclopedia of computational chemistry*, 1, 2002.

Mariani, V., Biasini, M., Barbato, A., and Schwede, T. lddt: a local superposition-free score for comparing protein structures and models using distance difference tests. *Bioinformatics*, 29(21):2722–2728, 2013.

Ramaraj, T., Angel, T., Dratz, E. A., Jesaitis, A. J., and Mumey, B. Antigen–antibody interface properties: Composition, residue interactions, and features of 53 non-redundant structures. *Biochimica et Biophysica Acta (BBA)-Proteins and Proteomics*, 1824(3):520–532, 2012.

Ramírez, D. and Caballero, J. Is it reliable to use common molecular docking methods for comparing the binding affinities of enantiomer pairs for their protein target? *International journal of molecular sciences*, 17(4):525, 2016.

Ramírez, D. and Caballero, J. Is it reliable to take the molecular docking top scoring position as the best solution without considering available structural data? *Molecules*, 23(5):1038, 2018.

Raybould, M. I., Marks, C., Krawczyk, K., Taddese, B., Nowak, J., Lewis, A. P., Bujotzek, A., Shi, J., and Deane, C. M. Five computational developability guidelines for therapeutic antibody profiling. *Proceedings of the National Academy of Sciences*, 116(10):4025–4030, 2019.

Ren, Z., Li, J., Ding, F., Zhou, Y., Ma, J., and Peng, J. Proximal exploration for model-guided protein sequence design. *bioRxiv*, 2022.

Ruffolo, J. A. and Gray, J. J. Fast, accurate antibody structure prediction from deep learning on massive set of natural antibodies. *Biophysical Journal*, 121(3):155a–156a, 2022.

Ruffolo, J. A., Gray, J. J., and Sulam, J. Deciphering antibody affinity maturation with language models and weakly supervised learning. *arXiv preprint arXiv:2112.07782*, 2021.

Saka, K., Kakuzaki, T., Metsugi, S., Kashiwagi, D., Yoshida, K., Wada, M., Tsunoda, H., and Teramoto, R. Antibody design using lstm based deep generative model from phage display library for affinity maturation. *Scientific reports*, 11(1):1–13, 2021.

Satorras, V. G., Hoogeboom, E., and Welling, M. E (n) equivariant graph neural networks. In *International conference on machine learning*, pp. 9323–9332. PMLR, 2021.

Schymkowitz, J., Borg, J., Stricher, F., Nys, R., Rousseau, F., and Serrano, L. The foldx web server: an online force field. *Nucleic acids research*, 33(suppl_2):W382–W388, 2005.

Shan, S., Luo, S., Yang, Z., Hong, J., Su, Y., Ding, F., Fu, L., Li, C., Chen, P., Ma, J., et al. Deep learning guided optimization of human antibody against sars-cov-2 variants with broad neutralization. *Proceedings of the National Academy of Sciences*, 119(11):e2122954119, 2022.

Steinegger, M. and Söding, J. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028, 2017.

Thomas, N., Smidt, T., Kearnes, S., Yang, L., Li, L., Kohlhoff, K., and Riley, P. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

Tiller, K. E. and Tessier, P. M. Advances in antibody design. *Annual review of biomedical engineering*, 17:191, 2015.

Xu, J. and Zhang, Y. How significant is a protein structure similarity with tm-score= 0.5? *Bioinformatics*, 26(7): 889–895, 2010.

Xu, M., Yu, L., Song, Y., Shi, C., Ermon, S., and Tang, J. Geodiff: A geometric diffusion model for molecular conformation generation. *arXiv preprint arXiv:2203.02923*, 2022.

Yan, Y., Tao, H., He, J., and Huang, S.-Y. The hdock server for integrated protein–protein docking. *Nature protocols*, 15(5):1829–1852, 2020.

Yuan, M., Wu, N. C., Zhu, X., Lee, C.-C. D., So, R. T., Lv, H., Mok, C. K., and Wilson, I. A. A highly conserved cryptic epitope in the receptor binding domains of sars-cov-2 and sars-cov. *Science*, 368(6491):630–633, 2020.

Zhang, Y. and Skolnick, J. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, 57(4):702–710, 2004.

A. Details of the Structural Initialization

Given an antibody, we denote the position number of the i-th residue as r_i. With the backbone template $\{Z_r \in \mathbb{R}^{3\times4} | r \in \mathbb{W}\}$ (§ 4.1) of the same numbering system, we initialize the backbone coordinates Z_{r_i} of the structure by linearly interpolating the residues between the well-conserved ones and extending the residues at two ends outwards. The above process can be formalized as follows:

$$Z_{r_i} = \begin{cases} Z_{r_i}, & r_i \in \mathbb{W}, \\ \frac{1}{q-p}[(i-p)Z_{r_q} + (q-i)Z_{r_p}], & r_i \notin \mathbb{W}, \exists p, q, \\ Z_{r_p} + (i-p)(Z_{r_p} - X_{p-1}^{bb}), & r_i \notin \mathbb{W}, \exists p, \nexists q, \\ Z_{r_q} + (q-i)(Z_{r_q} - X_{q+1}^{bb}), & r_i \notin \mathbb{W}, \nexists p, \exists q, \end{cases} \quad (1)$$

where p and q are defined as the indexes of the nearest conserved residues to the i-th residue when $r_i \notin \mathbb{P}$:

$$p = \max_k \{k | k < i, r_k \in \mathbb{W}\}, \quad (2)$$

$$q = \min_k \{k | k > i, r_k \in \mathbb{W}\}, \quad (3)$$

Z_{r_i} is then extended to $X_i^{(0)}$ by filling in the coordinates of the side-chain atoms with the coordinate of α-carbon.

B. Channel Attributes and Weights

Given a multi-channel coordinate $X \in \mathbb{R}^{3\times c}$, we assign it with a d-dimensional attribute matrix $A \in \mathbb{R}^{c\times d}$ and a set of weights $w \in \mathbb{R}^{c\times1}$. Each row vector of A is associated with the atom type and the atom position of the corresponding channel and the weights are decided by the residue type of the coordinate. Natural amino acids only incorporate four atom types (i.e. C, N, O, S), and each atom will be assigned a position code indicating the number of chemical bonds on the shortest path from it to the C_α (IUPAC et al., 1970). For example, Figure 6 illustrates the position code of each atom in the side chain of Tryptophan. The attribute vector for each channel is the sum of its atom type embedding and its position code embedding. For the unknown residues, we assign them with a maximum number of atom channels, where each channel is filled with a [MASK] atom type and a [MASK] atom position.
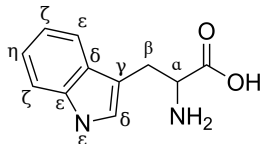

Figure 6. Position codes for atoms in the side-chain of Tryptophan.

Both attribute vectors and weights are learnable parameters in our model. For efficiency consideration, the value of d should be small because the dimension of the geometric relations $R_{ij} \in \mathbb{R}^{d\times d}$ (§ 4.3) is quadratic of d. In practice, we find that $d = 16$ is sufficient. Furthermore, we normalize the weight w with its L2-norm to avoid potential numerical instability.

C. Proof of Theorem 4.1

Theorem 4.1. *Given the initial epitope (along with the shadow paratope) $\{h_i, X_i\}_{i\in\mathcal{V}_E\cup\mathcal{V}_S}$ and the initialized antibody $\{h_i^{(0)}, X_i^{(0)}\}_{i\in\mathcal{V}_A}$, we compute the final prediction and docking by $\{p_i\}_{i\in\mathcal{V}_P}, \{\tilde{X}_i\}_{i\in\mathcal{V}_A} = dyMEAN\left(\{h_i, X_i\}_{i\in\mathcal{V}_E\cup\mathcal{V}_S}, \{h_i^{(0)}, X_i^{(0)}\}_{i\in\mathcal{V}_A}\right)$. We immediately have the conclusion that dyMEAN is E(3)-equivariant. Namely, for any transformations $g_1, g_2 \in E(3)$, we have $\{p_i\}_{i\in\mathcal{V}_P}, \{g_1 \cdot \tilde{X}_i\}_{i\in\mathcal{V}_A} = dyMEAN\left(\{h_i, g_1 \cdot X_i\}_{i\in\mathcal{V}_E\cup\mathcal{V}_S}, \{h_i^{(0)}, g_2 \cdot X_i^{(0)}\}_{i\in\mathcal{V}_A}\right)$ where $g \cdot X := QX + t$ for orthogonal transformation $Q \in O(3)$ and translation transformation $t \in \mathbb{R}^3$.*

We start by giving the definition of E(3)-invariance and E(3)-equivariance (Huang et al., 2022) as follows:

Definition C.1 (E(3)-equivariance). A function $\phi : \mathbb{X} \mapsto \mathbb{Y}$ is E(3)-equivariant if $\forall g \in \mathrm{E}(3)$ we have $\rho_{\mathbb{Y}}(g)\boldsymbol{y} = \phi(\rho_{\mathbb{X}}(g)\boldsymbol{x})$, where $\rho_{\mathbb{X}}$ and $\rho_{\mathbb{Y}}$ instantiate g in the input space $\mathbb{X}$ and the output space $\mathbb{Y}$. Specifically, ϕ is E(3)-invariant if $\rho_{\mathbb{Y}}(g) \equiv I$, where I is the identity transformation.

Before proceeding to the overall proof, we need to first present and prove several necessary lemmas below.

Lemma C.2. *For the geometric relation extractor T_R (§ 4.3), $\forall \boldsymbol{X}_i \in \mathbb{R}^{3 \times c_i}, \boldsymbol{X}_j \in \mathbb{R}^{3 \times c_j}$, suppose $\boldsymbol{R}_{ij} = T_R(\boldsymbol{X}_i, \boldsymbol{X}_j)$, then T_R is E(3)-invariant. Namely, $\forall g \in \mathrm{E}(3)$, we have $\boldsymbol{R}_{ij} = T_R(g \cdot \boldsymbol{X}_i, g \cdot \boldsymbol{X}_j)$, where $g \cdot \boldsymbol{X} := \boldsymbol{Q}\boldsymbol{X} + \boldsymbol{t}, \boldsymbol{Q} \in \mathrm{O}(3), \boldsymbol{t} \in \mathbb{R}^3$.*

Proof. $\forall \boldsymbol{X}_i \in \mathbb{R}^{3 \times c_i}, \boldsymbol{X}_j \in \mathbb{R}^{3 \times c_j}$, $\boldsymbol{R}_{ij}$ is obtained through Eq. 12. Consider the pair-wise channel distance matrix $\boldsymbol{D}_{ij}$, $\forall \boldsymbol{Q} \in \mathrm{O}(3), \boldsymbol{t} \in \mathbb{R}^3$, we have:

$$
\begin{aligned}
\boldsymbol{D}_{ij}(p, q) &= ||(\boldsymbol{Q}\boldsymbol{X}_i(:, p) + \boldsymbol{t}) - (\boldsymbol{Q}\boldsymbol{X}_j(:, q) + \boldsymbol{t})|| = ||\boldsymbol{Q}(\boldsymbol{X}_i(:, p) - \boldsymbol{X}_j(:, q))||, \\
&= \sqrt{[\boldsymbol{Q}(\boldsymbol{X}_i(:, p) - \boldsymbol{X}_j(:, q))]^\top [\boldsymbol{Q}(\boldsymbol{X}_i(:, p) - \boldsymbol{X}_j(:, q))]}, \\
&= \sqrt{(\boldsymbol{X}_i(:, p) - \boldsymbol{X}_j(:, q))^\top \boldsymbol{Q}^\top \boldsymbol{Q}(\boldsymbol{X}_i(:, p) - \boldsymbol{X}_j(:, q))}, \\
&= \sqrt{(\boldsymbol{X}_i(:, p) - \boldsymbol{X}_j(:, q))^\top (\boldsymbol{X}_i(:, p) - \boldsymbol{X}_j(:, q))}, \\
&= ||\boldsymbol{X}_i(:, p) - \boldsymbol{X}_j(:, q)||,
\end{aligned}
$$

With $\boldsymbol{A}_i, \boldsymbol{A}_j, \boldsymbol{w}_i, \boldsymbol{w}_j$ not affected by transformations on $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$, we can directly derive $\boldsymbol{R}_{ij} = T_R(\boldsymbol{Q}\boldsymbol{X}_i + \boldsymbol{t}, \boldsymbol{Q}\boldsymbol{X}_j + \boldsymbol{t}) = T_R(\boldsymbol{X}_i, \boldsymbol{X}_j)$, which concludes Lemma C.2. $\square$

Lemma C.3. *For the geometric message scaler T_S (§ 4.3), $\forall \boldsymbol{X} \in \mathbb{R}^{3 \times c}, \forall \boldsymbol{s} \in \mathbb{R}^C$, suppose $\boldsymbol{X}' = T_S(\boldsymbol{X}, \boldsymbol{s})$, then T_S is O(3)-equivariant. Namely, $\forall \boldsymbol{Q} \in \mathrm{O}(3)$, we have $\boldsymbol{Q}\boldsymbol{X}' = T_S(\boldsymbol{Q}\boldsymbol{X}, \boldsymbol{s})$.*

Proof. $\forall \boldsymbol{Q} \in \mathrm{O}(3), \boldsymbol{t} \in \mathbb{R}^3$, we have:

$$
T_S(\boldsymbol{Q}\boldsymbol{X}, \boldsymbol{s}) = (\boldsymbol{Q}\boldsymbol{X})\mathrm{diag}(\boldsymbol{s}') = \boldsymbol{Q}(\boldsymbol{X}\mathrm{diag}(\boldsymbol{s}') = \boldsymbol{Q}T_S(\boldsymbol{X}, \boldsymbol{s}) = \boldsymbol{Q}\boldsymbol{X}',
$$

which concludes Lemma C.3. $\square$

Lemma C.4. *Denote the AME (§ 4.3) as $\{\boldsymbol{h}'_i, \boldsymbol{X}'_i\}_{i \in \mathcal{V}_E \cup \mathcal{V}_S \cup \mathcal{V}_A} = AME(\{\boldsymbol{h}_i, \boldsymbol{X}_i\}_{i \in \mathcal{V}_E \cup \mathcal{V}_S \cup \mathcal{V}_A})$, then AME is independent E(3)-equivariant (Ganea et al., 2021) with respect to $\mathcal{V}_E \cup \mathcal{V}_S$ and $\mathcal{V}_A$. Namely, $\forall g_1, g_2 \in \mathrm{E}(3)$, we have $\{\boldsymbol{h}'_i, g_1 \cdot \boldsymbol{X}'_i\}_{i \in \mathcal{V}_E \cup \mathcal{V}_S} \cup \{\boldsymbol{h}'_i, g_2 \cdot \boldsymbol{X}'_i\}_{i \in \mathcal{V}_A} = AME(\{\boldsymbol{h}_i, g_1 \cdot \boldsymbol{X}_i\}_{i \in \mathcal{V}_E \cup \mathcal{V}_S} \cup \{\boldsymbol{h}_i, g_2 \cdot \boldsymbol{X}_i\}_{i \in \mathcal{V}_A})$.*

Proof. The key points to the proof of Lemma C.4 are the following two statements: (1) The information exchange between $\mathcal{V}_E \cup \mathcal{V}_S$ and $\mathcal{V}_A$ is E(3)-invariant; (2) The propagation process of Eq. 8-11 is E(3)-invariant on $\boldsymbol{h}$ and E(3)-equivariant on $\boldsymbol{X}$. If both (1) and (2) are right, then each layer of the AME satisfies independent E(3)-equivariance with respect to $\mathcal{V}_E \cup \mathcal{V}_S$ and $\mathcal{V}_A$, which easily leads to the correctness of Lemma C.4. Suppose (2) holds true, then the correctness of (1) is obvious because the information exchange between $\mathcal{V}_E \cup \mathcal{V}_S$ and $\mathcal{V}_A$ is conducted by the sharing of hidden states and topology between $\mathcal{V}_S$ and $\mathcal{V}_P$, both of which are E(3)-invariant. Thus the focus narrows down to the proof of (2).

$\forall g \in \mathrm{E}(3), g \cdot \boldsymbol{X} := \boldsymbol{Q}\boldsymbol{X} + \boldsymbol{t}, \boldsymbol{Q} \in \mathrm{O}(3), \boldsymbol{t} \in \mathbb{R}^3$, according to Lemma C.2, we have:

$$
\begin{aligned}
\boldsymbol{m}_{ij} &= \phi_m(\boldsymbol{h}_i^{(l)}, \boldsymbol{h}_j^{(l)}, \frac{T_R(\boldsymbol{X}_i^{(l)}, \boldsymbol{X}_j^{(l)})}{||T_R(\boldsymbol{X}_i^{(l)}, \boldsymbol{X}_j^{(l)})||_F + \epsilon}), \\
&= \phi_m(\boldsymbol{h}_i^{(l)}, \boldsymbol{h}_j^{(l)}, \frac{T_R(g \cdot \boldsymbol{X}_i^{(l)}, g \cdot \boldsymbol{X}_j^{(l)})}{||T_R(g \cdot \boldsymbol{X}_i^{(l)}, g \cdot \boldsymbol{X}_j^{(l)})||_F + \epsilon}),
\end{aligned}
$$

which reads that the computation of m_{ij} is E(3)-invariant. This directly leads to the E(3)-invariance of obtaining $h_i^{(l+1)}$ through Eq. 10. Next, according to Lemma C.3, we have:

$$\boldsymbol{QX}_{ij} = \boldsymbol{Q}T_S(\boldsymbol{X}_i^{(l)} - \frac{1}{c_j}\sum_{k=1}^{c_j}\boldsymbol{X}_j^{(l)}(:,k), \phi_x(\boldsymbol{m}_{ij})),$$

$$= T_S(\boldsymbol{Q}(\boldsymbol{X}_i^{(l)} - \frac{1}{c_j}\sum_{k=1}^{c_j}\boldsymbol{X}_j^{(l)}(:,k)), \phi_x(\boldsymbol{m}_{ij})),$$

$$= T_S(\boldsymbol{QX}_i^{(l)} - \frac{1}{c_j}\sum_{k=1}^{c_j}\boldsymbol{QX}_j^{(l)}(:,k), \phi_x(\boldsymbol{m}_{ij})),$$

$$= T_S(\boldsymbol{QX}_i^{(l)} + \boldsymbol{t} - \frac{1}{c_j}\sum_{k=1}^{c_j}(\boldsymbol{QX}_j^{(l)}(:,k) + \boldsymbol{t}), \phi_x(\boldsymbol{m}_{ij})),$$

$$= T_S(g \cdot \boldsymbol{X}_i^{(l)} - \frac{1}{c_j}\sum_{k=1}^{c_j}g \cdot \boldsymbol{X}_j^{(l)}(:,k), \phi_x(\boldsymbol{m}_{ij})),$$

which leads to the E(3)-equivariance of obtaining $\boldsymbol{X}_i^{(l+1)}$ through Eq. 11:

$$g \cdot \boldsymbol{X}_i^{(l+1)} = g \cdot (\boldsymbol{X}_i^{(l)} + \frac{1}{|\mathcal{N}(i)|}\sum_{j\in\mathcal{N}(i)}\boldsymbol{X}_{ij}),$$

$$= \boldsymbol{Q}(\boldsymbol{X}_i^{(l)} + \frac{1}{|\mathcal{N}(i)|}\sum_{j\in\mathcal{N}(i)}\boldsymbol{X}_{ij})) + \boldsymbol{t},$$

$$= \boldsymbol{QX}_i^{(l)} + \boldsymbol{t} + \boldsymbol{Q}\frac{1}{|\mathcal{N}(i)|}\sum_{j\in\mathcal{N}(i)}\boldsymbol{X}_{ij},$$

$$= g \cdot \boldsymbol{X}_i^{(l)} + \frac{1}{|\mathcal{N}(i)|}\sum_{j\in\mathcal{N}(i)}\boldsymbol{QX}_{ij}$$

Therefore, the propagation process of Eq. 8-11 is E(3)-invariant on $\boldsymbol{h}$ and E(3)-equivariant on $\boldsymbol{X}$, which concludes Lemma C.4. $\qquad\square$

Lemma C.5. *Denote the docking procedure in Eq. 17-18 as* $\{\tilde{\boldsymbol{X}}_i\}_{i\in\mathcal{V}_A} = \mathrm{Dock}(\{\boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_A}, \{\boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_S})$, *then it is E(3)-equivariant in terms of* $\mathcal{V}_S$. *Namely,* $\forall g_1, g_2 \in E(3)$, *we have* $\{g_1 \cdot \tilde{\boldsymbol{X}}_i\}_{i\in\mathcal{V}_A} = \mathrm{Dock}(\{g_2 \cdot \boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_A}, \{g_1 \cdot \boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_S})$.

Proof. Suppose $g \cdot X := \boldsymbol{QX} + \boldsymbol{t}$, where $\boldsymbol{Q}, \boldsymbol{t} = \mathrm{Kabsch}(\{\boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_P}, \{\boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_S})$. When we apply g_1 to $\mathcal{V}_S$ and g_2 to $\mathcal{V}_A$ ($\mathcal{V}_P \in \mathcal{V}_A$), the new Kabsch process can be interpreted as first exerting g_1^{-1} on $\mathcal{V}_S$ and g_2^{-1} on $\mathcal{V}_A$ to eliminate the transformations, then implementing the above-mentioned g on $\mathcal{V}_A$, and finally transforming $\mathcal{V}_A$ with g_1 to recover the transformation on $\mathcal{V}_S$. Therefore, for $g' \cdot X := \boldsymbol{Q}'X + \boldsymbol{t}'$, where $\boldsymbol{Q}', \boldsymbol{t}' = \mathrm{Kabsch}(\{g_2 \cdot \boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_P}, \{g_1 \cdot \boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_S})$, we have $g' = g_1 \cdot g \cdot g_2^{-1}$. Then it is easy to derive:

$$\{g' \cdot g_2 \cdot \boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_A} = \mathrm{Dock}(\{g_2 \cdot \boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_A}, \{g_1 \cdot \boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_S})$$

$$= \{g_1 \cdot g \cdot g_2^{-1} \cdot g_2 \cdot \boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_A} = \{g_1 \cdot g \cdot \boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_A}$$

$$= \{g_1 \cdot \tilde{\boldsymbol{X}}_i\}_{i\in\mathcal{V}_A},$$

which concludes Lemma C.5. $\qquad\square$

Ultimately we are ready to give the proof to Theorem 4.1 as follows:

Proof. $\forall g_1, g_2 \in E(3)$, according to Lemma C.4, each iteration in dyMEAN satisfies independent E(3)-equivariance with respect to $\mathcal{V}_E \cup \mathcal{V}_S$ and $\mathcal{V}_A$. Thus the transformed inputs $\{\boldsymbol{h}_i, g_1 \cdot \boldsymbol{X}_i\}_{i\in\mathcal{V}_E\cup\mathcal{V}_S}, \{\boldsymbol{h}_i^{(0)}, g_2 \cdot \boldsymbol{X}_i^{(0)}\}_{i\in\mathcal{V}_A}$ lead to transformed encoded results with independent E(3)-equivariance $\{\boldsymbol{h}_i^{(T)}, g_1 \cdot \boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_S}, \{\boldsymbol{h}_i^{(T)}, g_2 \cdot \boldsymbol{X}_i^{(T)}\}_{i\in\mathcal{V}_A}$. Next,

based on Lemma C.5, the docking procedure is E(3)-equivariant in terms of $\mathcal{V}_S$, thus we have $\{g_1 \cdot \tilde{X}_i\}_{i \in \mathcal{V}_A} = \text{Dock}(\{g_2 \cdot X_i^{(T)}\}_{i \in \mathcal{V}_A}, \{g_1 \cdot X_i^{(T)}\}_{i \in \mathcal{V}_S})$. Plus that p_i is obtained through Softmax on $h_i^{(T)}$, we can derive:

$$\{p_i\}_{i \in \mathcal{V}_P}, \{g_1 \cdot \tilde{X}_i\}_{i \in \mathcal{V}_A} = \text{dyMEAN}\left(\{h_i, g_1 \cdot X_i\}_{i \in \mathcal{V}_E \cup \mathcal{V}_S}, \{h_i^{(0)}, g_2 \cdot X_i^{(0)}\}_{i \in \mathcal{V}_A}\right),$$

which concludes Theorem 4.1. Also, the initialization of the shadow paratope conforms to the standard Gaussian distribution at the center of the epitope, which is isotropic (Xu et al., 2022) and thus does not interfere with the E(3)-equivariance. □

D. Adaption for Property Optimization

We can also adjust our method for optimizing the properties (e.g. affinity) of existing antibodies. (1) **Initialization with disturbance**. Since the structure of the existing antigen-antibody complex is known, we only need to disturb it with $\mathbf{Z} \sim \mathcal{N}(0, \mathbf{I})$ for initialization rather than the method in § 4.1. (2) **No shadow paratope**. The initialization also provides the relative position of the antibody to the epitope, therefore we can directly capture the interface geometry through interacting edges between epitope and antibody without the shadow paratope in § 4.2. A pair of residues is interacting if their distance is below a threshold δ (i.e. 6.6 Å according to Ramaraj et al. (2012)), therefore the interacting edge set is $\mathcal{E}_\mathcal{I} = \{(u, v) | d(u, v) \leq \delta, u \in \mathcal{V}_E, v \in \mathcal{V}_A\}$. The alternating message passing in the encoder AME can be merged into a single step on the entire complex $(\mathcal{V}_E \cup \mathcal{V}_A, \mathcal{E}_E \cup \mathcal{E}_A \cup \mathcal{E}_\mathcal{I})$. (3) **Partially masked sequence**. A common requirement of property optimization is to change the sequence as little as possible (Ren et al., 2022). To achieve flexibility in controlling the extent of modification, we only mask a random subset of the CDR residues $\mathbb{I}' \subseteq \mathbb{I}$ for each training step. Then we can set the upper bound of the number of modified residues by masking a fixed number of residues for generation.

Gradient Search Given a property scorer on complexes $f : \mathbb{G} \to \mathbb{R}$, we search for a favorable initialization that leads to a complex with the better property. we first utilize our trained model θ to generate a dataset $\mathcal{D} = \{(h_{\mathcal{G}_i}, f(\mathcal{G}_i)) | \mathcal{G}_i \sim \theta\}$, where $h_{\mathcal{G}_i}$ denotes the representation of the complex $\mathcal{G}_i$ obtained by averaging the hidden states of all nodes. The dataset $\mathcal{D}$ is then used to fit a predictor of f on the representation space of complexes: $\hat{f}_\theta : \mathbb{H} \to f[\mathbb{G}]$. Now given an existing complex $\mathcal{G}$ with CDRs partially masked, the mapping from the initial disturbance to the predicted score $\hat{f}_\theta \circ g_\theta(\mathcal{G}|\mathbf{Z})$ is differentiable, where g_θ sums up the process of generating a complex and then obtaining its representation. Suppose we want to maximize the property score, we can conduct gradient search on the initialization space by minimizing the target function:

$$\mathcal{L}(\mathbf{Z}) = -\hat{f}_\theta \circ g_\theta(\mathcal{G}|\mathbf{Z}) + D_{\text{KL}}(\mathbf{Z} \| \mathcal{N}(0, \mathbf{I})), \tag{4}$$

where the KL divergence (Kullback & Leibler, 1951) restricts the disturbance to the standard Gaussian distribution.

E. Huber Loss

The Huber loss (Huber, 1992) for robust supervision of the coordinates and bond lengths is defined as follows:

$$l(x, y) = \begin{cases} 0.5\,(x-y)^2, & if\ |x-y| < \delta, \\ \delta \cdot (|x-y| - 0.5 \cdot \delta), & else \end{cases} \tag{5}$$

When the deviation between x and y is below the threshold δ, l is equivalent to MSE loss, and when the deviation is above the threshold, l is equivalent to L1 loss. MSE loss provides smoothness near 0 but is sensitive to outliers, while the opposite holds true for L1 loss. By selecting a suitable loss according to the deviation, Huber loss combines the merits of MSE loss and L1 loss, thus exhibiting better numerical stability. We set $\delta = 1$ in our experiments following Kong et al. (2022).

F. Experiment Details

For the baselines, we adopt the hyperparameters and training procedure in their official releases since all the papers utilize SAbDab to form training sets of similar scale and distribution. We list the values of these hyperparameters as well as those of our dyMEAN in Table 6.

We train dyMEAN by Adam optimizer with the data-parallel framework of PyTorch on 2 GeForce RTX 2080 Ti GPUs. We set the initial $lr = 1 \times 10^{-3}$ and decay the learning rate exponentially to reach 1×10^{-4} at the last step. The batch size is 16, which is consistent across different tasks. It takes 200 epochs for dyMEAN to converge in the tasks of epitope-binding

CDR-H3 design and affinity optimization, while the number is 250 in the task of complex structure prediction. We notice that the learning of 1D sequences is faster than that of 3D structures, leading to overfitting the 1D sequences in CDR-H3 design. To bypass the problem, we unmask some paratope residues at the initial stage of training, and gradually transit to the ultimate setting where all the paratope residues are masked. Specifically, the ratio of unmasked paratope residues is initialized as 90% and anneals to 0% with a cosine schedule.

Table 6. Hyperparameters for the baselines and our dyMEAN.

hyperparameter	value	description
		HERN
hidden_size	256	Size of the hidden states in its hierachical message passing network (MPN).
num_rbf	16	Number of RBF kernels for distance embedding.
n_layers	4	Number of layers in the MPN.
k_neighbors	9	Number of neighbors for each node in the KNN graph.
		Diffab
hidden_size	128	Size of the hidden states in the MPN.
pair_size	64	Size of the residue-pair features.
n_layers	6	Number of layers in the MPN.
n_steps	100	Number of the diffusion steps.
		MEAN
embed_size	64	Size of the residue type embedding.
hidden_size	128	Size of the hidden states in the MPN
n_layers	3	Number of layers in the MPN
n_iter	3	Number of iterations in its progressive full-shot decoding.
k_neighbors	9	Number of neighbors for each node in the KNN graph.
		dyMEAN (ours)
embed_size	64	Size of the residue type embedding and the position number embedding.
hidden_size	128	Size of the hidden states in the MPN
n_layers	3	Number of layers in the MPN
n_iter	3	Number of iterations in the progressive full-shot decoding.
k_neighbors	9	Number of neighbors for each node in the KNN graph.
d	16	Size of the attribute vector of each channel (equal to the size of the atom type embedding and the atom position embedding).

Furthermore, we provide the definition of the epitope to HDock when using it for docking. To further enhance its docking performance, we generate 100 docked samples for each antibody and calculate the 48 residues closest to each antibody. We compare these residues with the given epitope and select the candidate with the top-1 coverage as the final result.

G. Space Complexity Analysis

We emphasize the spatial efficiency of our dyMEAN compared to HERN (Jin et al., 2022), which also models the side chains in addition to backbones but is limited to the paratope (*i.e.*, CDR-H3), by two aspects: initialization and encoding. We denote the number of residues in the epitope, the paratope, and the antibody by N_E, N_P, and N_A.

Initialization It is obvious that our structural initialization (§ 4.1) has a complexity linear to the number of residues in the shadow paratope and the antibody, which is $O(N_P + N_A)$. HERN initializes the coordinates of the antibody via eigenvalue decomposition of the residue-level pair-wise distance matrix of the complex, thus having a complexity of $O((N_E + N_P)^2)$. Since N_A is much larger than N_P, the quadratic complexity largely impedes HERN's scaling from modeling paratope only to modeling the entire antibody (*i.e.*, replacing N_P with N_A).

Encoding The space complexity of GNN-based encoders is dominated by the scheme of edge-wise message passing. We denote the maximum number of neighbors of each node by K, and the maximum number of atoms in a single residue (*i.e.*, maximum channel size) by C. For our AME (§ 4.3), the major influence is the geometric relation extractor (Eq. 12) with a complexity of $O(K(N_E + N_P + N_A)(2dC + 2C + C^2)) = O(K(N_E + N_P + N_A)C(2d + 2 + C))$, where d is

the dimension of the attribute vector. HERN adopts a hierarchical encoder, which first implements EGNN (Satorras et al., 2021) on the residue-level graph and the atom-level graph sequentially, then updates the coordinates with inter-C_α terms and intra-residue terms. Since the atom-level graph is much larger than its residue-level counterpart, the dominant part is the atom-level message passing with a complexity of $O(K(N_E + N_P)CH)$, where H denotes the hidden size. Scaling HERN from N_P to N_A, we have $N_E + N_P + N_A \approx N_E + N_A$ but $H >> 2d + 2 + C$ because d is set to a small number and $C = 14$ in the dataset, which reveals the superiority of AME in efficiency over the hierarchical encoder.

Our attempts to scale HERN to the entire antibody bring no success and exhibit unrealistic GPU requirements, which we attribute to the high complexity of its initialization, hierarchical encoder, and autoregressive refinement (Kong et al., 2022).

H. Side-Chain Dihedral Angles

To analyze whether our model generates realistic dihedral angles in the side chains, we draw the distribution of $\chi_1, \chi_2, \chi_3, \chi_4$ with the generated structures and the reference structures[2]. We display the overall distribution in figure 7 and separated distribution of different amino acids in figure 8. Both figure 7 and figure 8 show that generally the generated dihedral angles conforms to the reference distribution. However, we also identify from the fine-grained figure 8 that the generated distributions are smoother than the reference ones, indicating possible minor deviations on the angles under certain circumstances. Therefore, in some practical applications, relaxing methods like OpenMM (Eastman et al., 2017) are still needed for post-process.
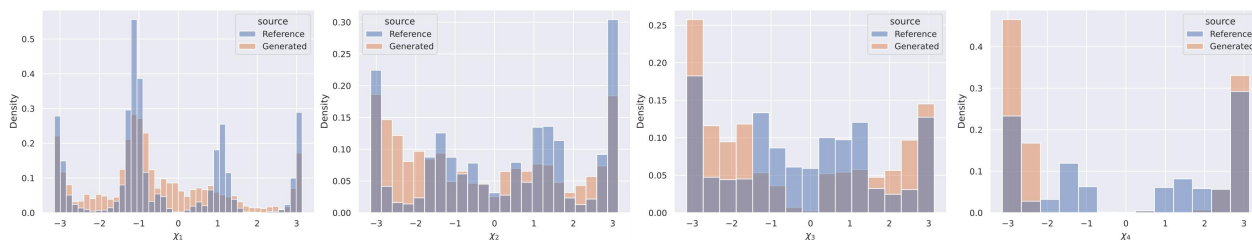

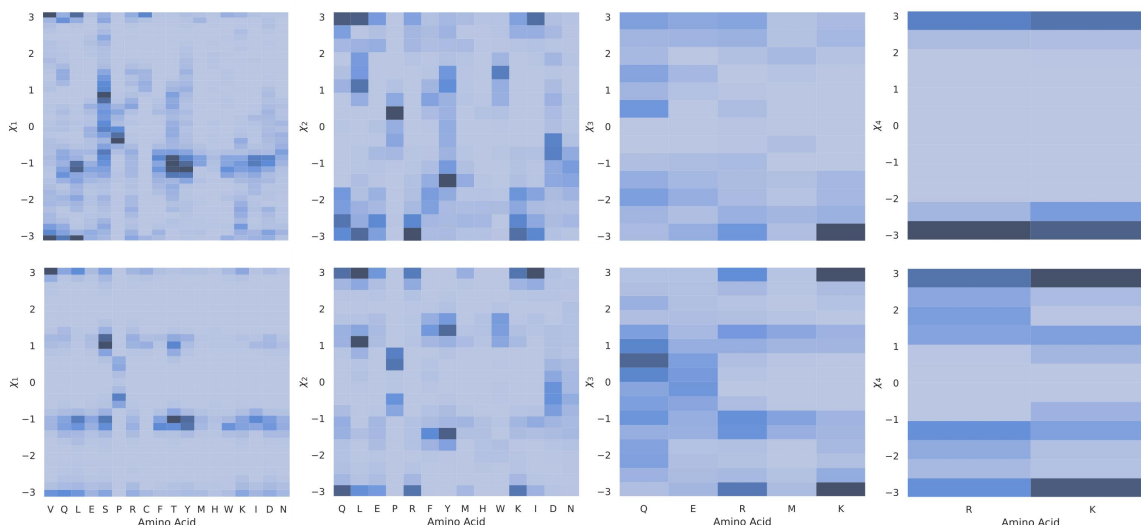*Figure 7.* The overall distributions of 4 dihedral angles in the side chains.

Figure 8. The distribution of 4 dihedral angles in the side chains categorized by amino acids. The first row and the second row display the distributions from the generated structures and the reference structures, respectively.

---

[2]We use the following definitions of the dihedral angles: http://www.mlb.co.jp/linux/science/garlic/doc/commands/dihedrals.html

I. Threshold for Defining Conserved Residues

We analyze the influence of the threshold for defining the conserved residues (§ 4.1) by displaying the variations in the number of conserved residues and the average RMSD of the antibodies in the dataset to the conserved template. According to Figure 9, with the decrease of the threshold, the number of conserved residues gradually increases and becomes stable at 90% threshold, which is expected. In contrast, the RMSD curve shows a surge at 92% threshold. Hence, to balance these two factors, we think it is better to set the threshold between 93% and 96%, where the RMSD remains low and the number of conserved residues is not too small.
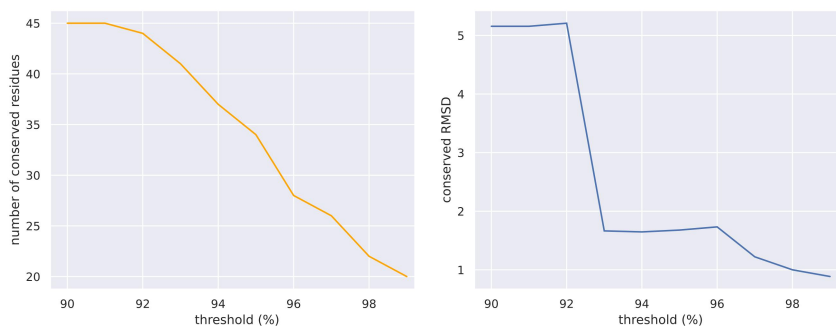

Figure 9. Number of conserved residues (left) and average RMSD of the antibodies in the dataset to the conserved template (right) with respect to different thresholds for defining the conserved residues.

We further conducted experiments with templates from 90% and 99% threshold, and show the results in Table 7. It is observed that our model is robust with the templates, but still, using 95% as threshold generally achieves more favorable results compared to the other choices.

Table 7. Performance of dyMEAN with conserved templates defined by different thresholds.

Threshold	Generation			Docking		
	AAR↑	TMscore↑	LDDT↑	CAAR↑	RMSD↓	DockQ↑
CDR-H3 Design						
90%	40.99%	0.9722	0.8365	26.04%	**7.88**	**0.415**
95%	**43.65%**	0.9726	**0.8454**	28.11%	8.11	0.409
99%	43.32%	**0.9730**	0.8438	27.59%	9.30	0.405
Complex Structure Prediction						
90%	-	0.9703	0.8597	-	9.34	**0.459**
95%	-	**0.9731**	**0.8673**	-	**9.05**	0.452
99%	-	0.9712	0.8601	-	9.70	0.443

J. Selection of the Shadow Paratope in Docking

Table 8. Results on complex structure prediction using both CDR-H3 and CDR-L3 as the shadow paratope.

Shadow Paratope	TMscore↑	lDDT↑	RMSD↓	DockQ↑
H3	0.9731	0.8673	9.05	0.452
H3 + L3	0.9585	0.8248	10.80	0.397

In this work, we generally use CDR-H3 since the interacting residues mainly come from it (Akbar et al., 2021). In practice, we find that using CDR-H3 only is generally sufficient for docking. Also, it is easy to extend the shadow paratope to contain other CDRs by our implementation. We conduct an experiment using both CDR-H3 and CDR-L3 as the shadow paratope for the docking experiment and the results are in Table 8. Additional inclusion of CDR-L3 leads to a slight performance

drop in all metrics. This is possibly because the two CDRs are separated spatially in the structure, and identifying their relative positions brings up additional complexity to the problem.

K. Affinity with FoldX

We provide the evaluation on affinity optimization (§ 5.3) with FoldX (Schymkowitz et al., 2005) as the criterion in Table 9. We first use OpenMM (Eastman et al., 2017) to relax the generated structures, then use FoldX to minimize the energy. Finally, we use FoldX to calculate the interface energy (ΔG) of both the wild type and the mutant to obtain the $\Delta\Delta G$.

Table 9. Average $\Delta\Delta G$ (kcal/mol) and average number of changed residues (ΔL). dyMEAN-n denotes the restricted version allowing at most n changed residues, and dyMEAN itself changes n residues with n is sampled from $[1, N]$ at each generation.

Method	$\Delta\Delta G\downarrow$	$\Delta L\downarrow$	Method	$\Delta\Delta G\downarrow$	$\Delta L\downarrow$
Diffab ($t=1$)	-0.72	1.08	MEAN	-5.84	5.09
Diffab ($t=2$)	-0.77	1.17	dyMEAN-1	-7.95	0.94
Diffab ($t=4$)	-0.42	1.10	dyMEAN-2	**-8.36**	1.27
Diffab ($t=8$)	0.07	1.42	dyMEAN-4	-8.33	2.08
Diffab ($t=16$)	0.74	2.67	dyMEAN-8	-7.89	4.37
Diffab ($t=32$)	1.77	6.40	dyMEAN	-8.10	2.76

L. Trial on General Proteins

We have further conducted experimental validation on the CATH dataset. We select the complexes from the CATH dataset and divide each of them into a receptor and a ligand. We identify the interacting residues of the ligands and mask them for generation, with the rest of the ligand as the "framework". Two types of settings are applied for validation. In the "e2e" setting, neither the structure nor the docking position of the framework is provided, resembling the setting in this paper. In the "inpainting" setting, both the structure and the docking position of the framework are provided, thus the model only focuses on filling in the interacting residues and adjusting the framework structure according to the docking position, imitating the setting in previous works (e.g. DiffAb, MEAN). The total number of valid data is 6883. In the e2e setting, we use ESMFold (Lin et al., 2023) to provide the initial structures in place of the method in § 4.1. Results in Table 10 illustrate that dyMEAN achieves promising performance in inpainting the protein complexes. Nevertheless, the performance on the e2e setting reveals greater challenges in designing binding interfaces without prior knowledge on the binding positions. Augmenting data or upgrading models are still urgent needs in this domain.

Table 10. Evaluation on general proteins in the CATH dataset.

Task	AAR$\uparrow$	TMscore$\uparrow$	lDDT$\uparrow$	RMSD$\downarrow$	DockQ$\uparrow$
e2e	16.02%	0.7678	0.6579	11.28	0.188
inpainting	51.79%	0.9708	0.8481	0.66	0.916

M. Samples

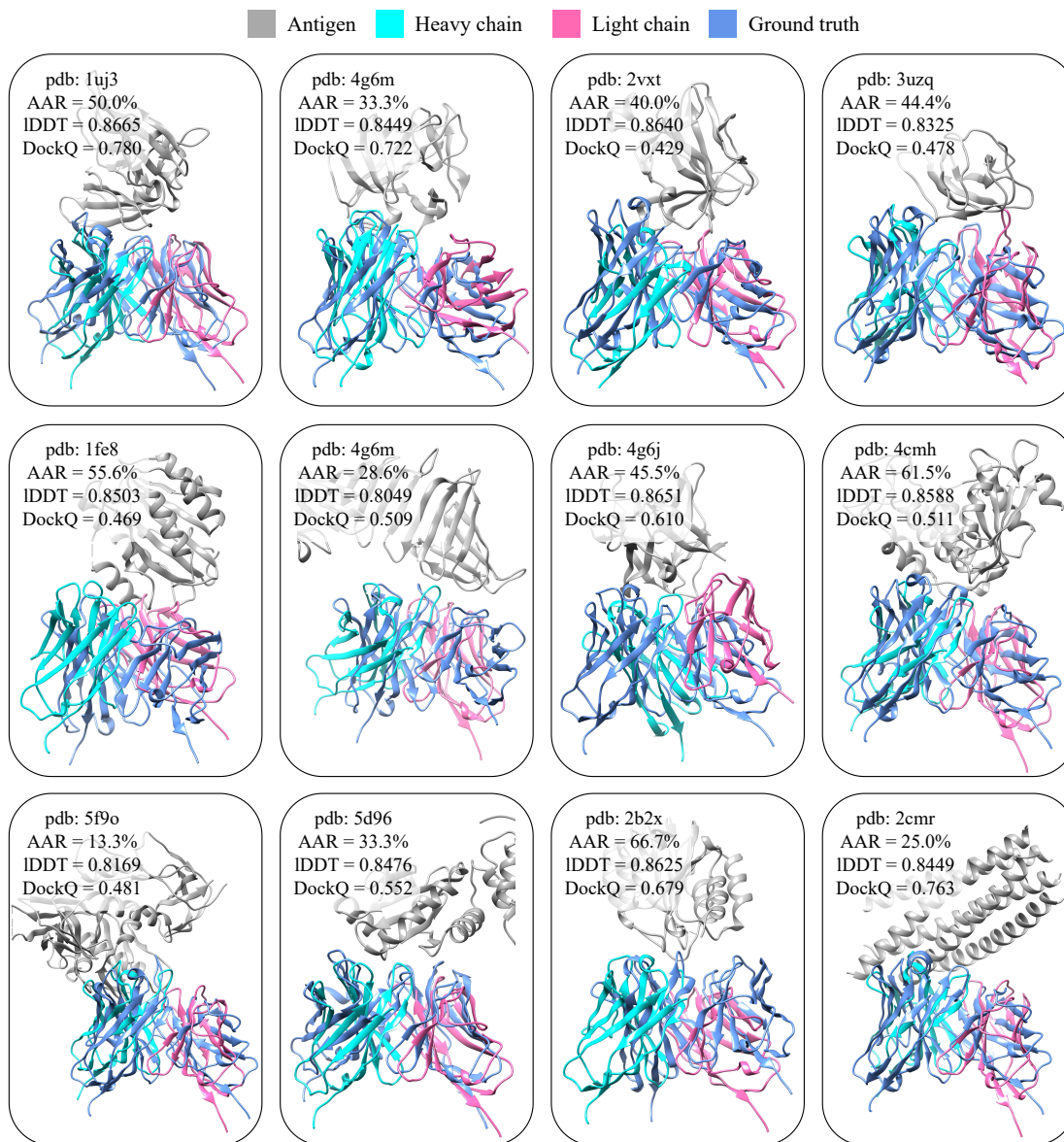We provide more samples of generated antibodies from epitope-binding CDR-H3 design (§ 5.1) in Figure 10.

Figure 10. Samples of generated antibodies.