AbODE: Ab initio antibody design using conjoined ODEs

Yogesh Verma [1] **Markus Heinonen** [1] **Vikas Garg** [1 2]

Abstract

Antibodies are Y-shaped proteins that neutralize pathogens and constitute the core of our adaptive immune system. *De novo* generation of new antibodies that target specific *antigens* holds the key to accelerating vaccine discovery. However, this co-design of the amino acid sequence and the 3D structure subsumes and accentuates, some central challenges from multiple tasks, including protein folding, inverse folding, and docking. We strive to surmount these challenges with a new generative model `AbODE` that extends graph PDEs to accommodate both contextual information and external interactions. Unlike existing approaches, `AbODE` uses a single round of full-shot decoding, and elicits continuous differential attention that encapsulates, and evolves with, latent interactions within the antibody as well as those involving the antigen. We unravel fundamental connections between `AbODE` and temporal networks as well as graph-matching networks. The proposed model significantly outperforms existing methods on standard metrics across benchmarks.

1. Introduction

Machine learning methods have recently enabled exciting developments for computational drug design, including tasks such as protein folding (Jumper et al., 2021), sequence design or inverse folding (Ingraham et al., 2019), and docking (Ganea et al., 2021).

We focus on the problem of *antibody design*. Antibodies, the versatile Y-shaped proteins that guard against pathogens, are essential to our adaptive immune mechanism. Typically, an antibody binds to a specific part of the pathogen, namely, the *antigen*. Each antibody recognizes a unique antigen, and the so-called Complementarity Determining Regions (CDRs) at the tip of the antibody determine this specificity (Figure 1). Thus, automating the design of antibodies against specific pathogens (e.g., the SARS-CoV-2 virus) can revolutionize drug discovery (Pinto et al., 2020).

We aim to co-design the CDR sequence and structure, conditioned on an antigen. While recent generative methods for protein sequence design have been successful (Ingraham et al., 2019), they crucially utilize that the long-term dependencies in sequence are local in the 3D space. However, the CDR structures are seldom known a priori, limiting the scope of such approaches. In principle, one could segregate the design of sequence from the structure. Indeed, once a CDR sequence is generated, methods such as AlphaFold(Jumper et al., 2021) can be employed to estimate the 3D structure of the CDR. However, generating sequences without conditioning on the structure (Alley et al., 2019) produces sub-optimal sequences.

Initial approaches for antibody design (Pantazes & Maranas, 2010; Li et al., 2014) relied on hand-crafted energy functions that entailed expensive simulation. Going beyond 1D sequence prediction (Alley et al., 2019; Shin et al., 2021a; Akbar et al., 2022), recent generative methods co-design structure and sequence (Jin et al., 2022b) and can incorporate information about antigens directly in the model (Jin et al., 2022a). However, the autoregressive scheme adopted by (Jin et al., 2022a) is susceptible to issues such as vanishing or exploding gradients during training, slow generation, and accumulation of errors during inference. Kong et al. (2023) advocate multiple *full-shot* rounds to address this issue; however, segregating context (intra-antibody) from external interactions (antibody-antigen) precludes joint optimization and may result in sub-optimality.

We model the antibody-antigen complex as a joint 3D graph with heterogeneous edges. Different from all prior works, this perspective allows us to formulate a coupled neural ODE system over the antibody while simultaneously accounting for the antigen. Specifically, we associate local densities (one per antibody node) progressively refined toward globally aligned densities based on simultaneous feedback from the antigen and the (other) antibody nodes. The 3D coordinates and the node labels for the antibody can then be sampled after a few rounds in *one-shot*, i.e., all at once. Thus, the entire procedure is efficient and end-to-end trainable.

[1]Department of Computer Science, Aalto University, Finland [2]YaiYai Ltd. Correspondence to: Yogesh Verma <yogesh.verma@aalto.fi>.

Accepted to ICML workshop on Structured Probabilistic Inference Generative Modeling

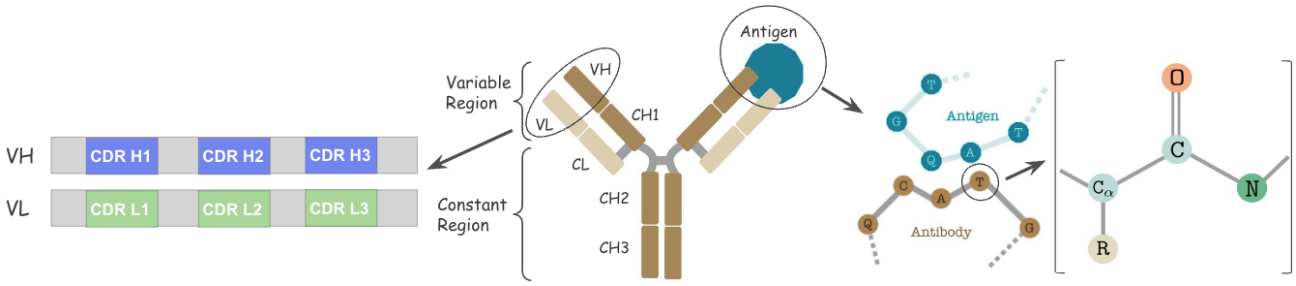

Figure 1: Schematic showing the structure of a residue (amino acid), where the backbone atoms we use are N, C_α and C (**right**) and the structure of the antibody (**left**) which is Y-shaped showing the VH/VL sequences and binding to the antigen, and we focus on CDRs of the variable domain in the heavy chain (VH).

We show how invariance can be built into the proposed method `AbODE` that accounts for rotations and other symmetries. `AbODE` establishes a new state-of-the-art (SOTA) for antibody design across several benchmarks. Interestingly, it shares connections with methods for equivariant molecular generation and docking: ModFlow (Verma et al., 2022) and IEGMN (Ganea et al., 2021). While ModFlow can be recovered as a particular case of the `AbODE` , IEGMN may be interpreted as a discrete analog of `AbODE`. These similarities reaffirm the kinship of different computational drug design tasks; conversely, they suggest the broader applicability of neural PDEs as effective tools for these tasks.

2. Antibody sequence and structure co-design

An antibody (Ab) is a Y-shaped protein (Fig. 1) that identifies antigens and stimulates an immunological response. An antibody consists of a constant domain and a symmetric variable region divided into heavy (H) and light (L) chains (Kuroda et al., 2012). The surface of the antibody contains three complementarity-determining regions (CDRs), which act as the primary binding determinant. CDR-H3 makes up most of the binding affinity (Fischman & Ofran, 2018). The non-CDR regions are highly preserved (Kuroda et al., 2012); thus, it is common to formulate antibody design as a CDR design problem (Shin et al., 2021b). We view the antibody-antigen complex as a joint graph with interactions between nodes across the binding. We co-model both the sequence and the 3D conformation of the CDR regions with a graph PDE and apply our method to antigen-specific and unconditional antibody design tasks.

2.1. The antibody-antigen graph

We define the antigen-antibody complex as a 3D graph $G = (V, E, X)$, with antibody `Ab` and antigen `Ag` amino-acid residues as vertices $V = (V_{\texttt{Ab}}, V_{\texttt{Ag}})$, coordinates $X = (X_{\texttt{Ab}}, X_{\texttt{Ag}})$ and edges $E = (E_{\texttt{Ab}}, E_{\texttt{Ab-Ag}})$ within the antibody as well as between the antibody and the antigen. Each vertex $v \in \mathcal{A}\{\texttt{Arg}, \texttt{His}, \ldots\}$ is one of 20 amino acids. We treat the labels with a Categorical distribution, such that the label features $\mathbf{a}_i \in \mathrm{R}^{20}$ represent the unnormalized amino acid probabilities. We also represent each residue by the cartesian 3D coordinates of its three backbone atoms $\{N, C_\alpha, C\}$ (see Fig. 1). For the i^{th} residue $\mathbf{x}_i$ we compute its spatial features $\mathbf{s}_i = (r_i, \alpha_i, \gamma_i)$, where

$$r_i = \|\mathbf{u}_i\|, \quad \mathbf{u}_i = \mathbf{x}_{i+1} - \mathbf{x}_i \tag{1}$$

$$\alpha_i = \cos^{-1}\left(\frac{\langle \mathbf{u}_i, \mathbf{u}_{i-1}\rangle}{\|\mathbf{u}_i\| \cdot \|\mathbf{u}_{i-1}\|}\right) \tag{2}$$

$$\gamma_i = \cos^{-1}\left(\frac{\langle \mathbf{u}_i, \mathbf{n}_i\rangle}{\|\mathbf{u}_i\| \cdot \|\mathbf{n}_i\|}\right), \quad \mathbf{n}_i = \mathbf{u}_i \times \mathbf{u}_{i-1}. \tag{3}$$

Here, r_i is the distance between consecutive residues x_i and x_{i+1}, α_i is the co-angle of residue i w.r.t previous and next residue, γ_i is the azimuthal angle of i's local plane, and $\mathbf{n}_i$ is the normal vector. The full residue state $\mathbf{z}_i = [\mathbf{a}_i, \mathbf{s}_i]$ concatenates the label features $\mathbf{a}_i$ and the spatial features $\mathbf{s}_i$.

Interactions To capture the interactions, we define edges $E_{\texttt{Ab}}$ between all antibody residues and edges $E_{\texttt{Ab-Ag}}$ between all antibody and antigen residues (See Figure 2). We also define edge features between all nodes i and j,

$$\mathbf{e}_{ij} = (\Delta \mathbf{z}_{ij}, i - j, \mathrm{RBF}\left(\|\mathbf{s}_i - \mathbf{s}_j\|\right). \tag{4}$$

$$\mathcal{O}_i^\top \frac{s_{i,\alpha} - s_{j,\alpha}}{\|s_{i,\alpha} - s_{j,\alpha}\|}, \mathcal{O}_i^\top \mathcal{O}_j, k_{ij}). \tag{5}$$

These include state differences $\Delta \mathbf{z}_{ij} = \{\Delta \mathbf{a}_{ij}, \Delta \mathbf{s}_{ij}\}$, backbone distance $i - j$, and spatial distance $\texttt{RBF}(||\mathbf{s}_i \mathbf{s}_j||)$ (here, RBF is the standard radius basis function kernel). The fourth term encodes directional embedding in the relative direction of j in the local coordinate frame $\mathcal{O}_i$ (Ingraham et al., 2019), and the $\mathcal{O}_i^T \mathcal{O}_j$ describes the orientation encoding of the node i with node j (See Appendix A.1 for details). Finally, we encode within-antibody edges with $k = 1$ and antibody-antigen edges with $k = 2$.

Task formulation Given a three-dimensional antibody or antibody-antigen graph, we aim to jointly learn a PDE to generate an amino acid sequence and the corresponding 3D conformation.

2.2. Conjoined system of ODEs

We propose to model the distribution of antibody-antigen complexes by a differential graph flow $\mathbf{z}(t)$ over time $t \in \mathrm{R}_+$. We initialize the initial state $\mathbf{z}(0)$ to a uniform categorical vector(Jin et al., 2022b; Kong et al., 2023). Coordinates are initialized with the even distribution between the residue right before CDRs and the one right after CDRs following (Kong et al., 2023), and we learn $\frac{d\mathbf{z}(t)}{dt}$ that maps to the end state $\mathbf{z}(T)$ that matches data. We begin by assuming an ODE system $\{\mathbf{z}_i(t)\}$ over time $t \in \mathrm{R}_+$, where node the time evolution of node i is an ODE

$$\dot{\mathbf{z}}_i(t) = \frac{\partial \mathbf{z}_i(t)}{\partial t} = f_\psi(t, \mathbf{z}_i(t), \mathbf{z}_{N(i)}(t), \{\mathbf{e}_{ij}(t)\}_j) \quad (6)$$

where $N(i) = \{j : (i,j) \in E\}$ indexes the neighbors of node i, and the function f parameterized by ψ is our main learning goal. The differentials form a coupled ODE system

$$\dot{\mathbf{z}}(t) = \begin{pmatrix} \dot{\mathbf{z}}_1(t) \\ \vdots \\ \dot{\mathbf{z}}_M(t) \end{pmatrix} \quad (7)$$

$$= \begin{pmatrix} f_\psi(t, \mathbf{z}_1(t), \mathbf{z}_{N(1)}(t), \{\mathbf{e}_{1j}(t)\}_j) \\ \vdots \\ f_\psi(t, \mathbf{z}_M(t), \mathbf{z}_{N(M)}(t), \{\mathbf{e}_{Mj}(t)\}_j)) \end{pmatrix} \quad (8)$$

$$\mathbf{z}(T) = \mathbf{z}(0) + \int_0^T \dot{\mathbf{z}}(t)dt \,. \quad (9)$$

where M is the number of nodes. Interestingly, it turns out that the PDE obtained using a recently proposed method for molecular generation can be recovered as a particular case of 7, when all the edges are set to be of the same type.

Proposition 1 : *ModFlow (Verma et al., 2022) can be seen as a special case of* `AbODE` *in an unconditional setting. This can be achieved by setting $k_{ij} = 1$ for every e_{ij}*

2.3. Attention-based differential

We capture the interactions between the antigen and antibody residues with graph attention (Shi et al., 2020)

$$\alpha_{ij} = \mathrm{softmax}\left(\frac{(\mathbf{W}_3\mathbf{z}_i)^\top (\mathbf{W}_4\mathbf{z}_j + \mathbf{W}_6\mathbf{e}_{ij})}{\sqrt{d}}\right) \quad (10)$$

$$\mathbf{z}'_i = \mathbf{W}_1\mathbf{z}_i + \sum_{j \in N(i)} \alpha_{ij} (\mathbf{W}_2\mathbf{z}_j + \mathbf{W}_6\mathbf{e}_{ij}) \quad (11)$$

where $\mathbf{W}_1, \ldots, \mathbf{W}_6$ are weight parameters and d is the head size. The α's are the attention coefficients corresponding to within and across edges, which are used to update the node feature $\mathbf{z}_i$. Interestingly, our method also shares similarities with the Independent E(3)-Equivariant Graph Matching Networks (IEGMNs) for docking (Ganea et al., 2021).

Proposition 2 : `AbODE` *can be cast as Independent E(3)-Equivariant Graph Matching Networks (IEGMN) (Ganea et al., 2021)). The operations are listed in Table 3 (See Appendix A.2 for more details).*

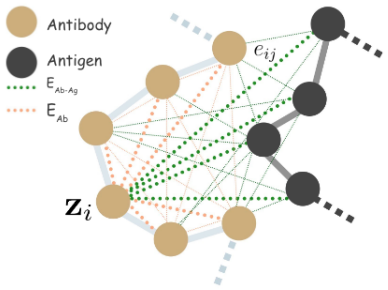

Figure 2: Schematic graph construction for the antigen-antibody complex with internal edges $E_{\mathtt{Ab}}$ and external edges $E_{\mathtt{Ab-Ag}}$. In the unconditional setting (i.e., the antigen is not specified), this reduces to an antibody graph

2.4. Training Objective

We optimize for the data fit of the generated states $z(T)$ by two components: one for the sequence and another for the structure

$$\mathcal{L} = \mathcal{L}_{\mathtt{seq}} + \mathcal{L}_{\mathtt{structure}} \quad (12)$$

The sequence loss is quantified in terms of the cross-entropy between the true label $\mathbf{a}_{ni}^{\mathtt{true}}$ and the label distribution $\mathbf{a}_{ni}$predicted by the model, i.e.,

$$\mathcal{L}_{\mathtt{seq}} == \frac{1}{N}\sum_{n=1}^{N}\frac{1}{M}\sum_{i=1}^{M_i} \mathrm{CE}\left(\mathbf{a}_{ni}^{\mathtt{true}}, \mathbf{a}_{ni}\right) \quad (13)$$

where n indexes the N datapoints and i indexes the M_i residues. The structure loss is computed based on the fit to the data sample in terms of the angles and radii:

$$\mathcal{L}_{\mathtt{structure}} = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{M}\sum_{i=1}^{M_i} \lambda\left(\mathcal{L}^{ni}_{\mathtt{angle}} + \mathcal{L}^{ni}_{\mathtt{radius}}\right). \quad (14)$$

For each residue angle pair (α, γ) we compute the negative log of the von-Mises likelihood

$$\mathcal{L}^{ni}_{\mathtt{angle}} = \sum_{k}^{\{\mathrm{C}_\alpha,\mathrm{C},\mathrm{N}\}} \sum_{\theta \in \{\alpha,\gamma\}} \log \mathcal{M}\left(\theta^n_{ik} \mid \theta^{n,\,\mathtt{true}}_{ik}, \kappa\right) \quad (15)$$

Table 1: **Top**: Unconditional sequence and structure benchmark. Baselines are from Jin et al. (2022b). **Bottom**: Antigen-conditional sequence and structure benchmark. Baselines are from Kong et al. (2023).

	CDR-H1		CDR-H2		CDR-H3	
Method	PPL (↓)	RMSD (↓)	PPL (↓)	RMSD (↓)	PPL (↓)	RMSD (↓)
LSTM	6.79	(N/A)	7.21	(N/A)	9.70	(N/A)
AR-GNN	6.44	2.97	6.86	2.27	9.44	3.63
RefineGNN	6.09	1.18	6.58	0.87	8.38	2.50
AbODE	**4.25** ± 0.46	**0.73** ± 0.15	**4.32** ± 0.32	**0.63** ± 0.19	**6.35** ± 0.29	**2.01** ± 0.13

	CDR-H1		CDR-H2		CDR-H3	
Method	AAR % (↑)	RMSD (↓)	AAR % (↑)	RMSD (↓)	AAR % (↑)	RMSD (↓)
LSTM	40.98 ± 5.20	(N/A)	28.50 ± 1.55	(N/A)	15.69 ± 0.91	(N/A)
C-LSTM	40.93 ± 5.41	(N/A)	29.24 ± 1.08	(N/A)	15.48 ± 1.17	(N/A)
RefineGNN	39.40 ± 5.56	3.22 ± 0.29	37.06 ± 3.09	3.64 ± 0.40	21.13 ± 1.59	6.00 ± 0.55
C-RefineGNN	33.19 ± 2.99	3.25 ± 0.40	33.53 ± 3.23	3.69 ± 0.56	18.88 ± 1.37	6.22 ± 0.59
MEAN	58.29 ± 7.27	0.98 ± 0.16	47.15 ± 3.09	0.95 ± 0.05	36.38 ± 3.08	2.21 ± 0.16
AbODE	**70.5** ± 1.14	**0.65** ± 0.1	**55.7** ± 1.45	**0.73** ± 0.14	**39.8** ± 1.17	**1.73** ± 0.11

where κ is a scale parameter, and k is atom index. The von Mises distribution can be interpreted as a Gaussian distribution over the domain of angles. On the other hand, the radius loss is the negative log of a Gaussian distance.

$$\mathcal{L}^{ni}_{\texttt{radius}} = \sum_{k}^{\{c_\alpha,C,N\}} \log \mathcal{N}\left(r^{n}_{ik} \mid r^{n,\,\text{true}}_{ik}, \sigma^2_r\right) \tag{16}$$

where σ^2_r is the radius variance. Here λ is the polar loss weight, set to $\lambda = 0.8$. We set $\kappa = 10$, $\sigma^2_r = 0.1$ to prefer narrow likelihoods for accurate structure prediction.

2.5. Sequence and structure prediction

Given the antibody or antigen-antibody complex, we generate an antibody sequence and the structure by solving the system of ODEs for time T to obtain $\mathbf{z}(T) = [\mathbf{a}(T), \mathbf{s}(T)]$. We transform the label features $\mathbf{a}(T)$ into Categorical amino acid probabilities $\mathbf{p}$ using the softmax operator. We pick the most probable amino acid per node.

3. Experiments

Tasks We benchmark AbODE on unconditional, conditional antibody sequence and structure generation against ground truth structures in the Structural Antibody Database SAbDab (Dunbar et al., 2014), and its ability to generate antigen-conditioned antibody sequences and structures from SAbDab. Moreover, we also evaluate the functional validity of the generated antibodies by considering various properties. Finally, we evaluate our model on the task of designing CDR- H3 over 60 manually selected diverse complexes (Adolf-Bryfogle et al., 2018).

Data We obtained the antibody sequences and structure from Structural Antibody Database (SAbDab) (Dunbar et al., 2014) and removed the illegal data points, renumbering them to the IMGT scheme (Lefranc et al., 2003). We followed a similar strategy to Jin et al. (2022b); Kong et al. (2023), where we focused on generating heavy chain CDRs, and by clustering the CDR sequences via MMseq2 (Steinegger & Söding, 2017) with 40% sequence identity. We then randomly split the clusters into training, validation, and test sets with an 8:1:1 ratio.

Metrics We evaluate our method on perplexity (PPL) and root mean square deviation (RMSD) between the predicted structures and the ground truth structures on the test data for the unconditioned sequence design task. Additionally, we report Amino Acid Recovery (AAR) for the antigen-conditioned sequence and structure design task. AAR is the overlapping rate between the predicted 1D sequences and the ground truth. We report the results for all the CDR-H regions. We calculate the RMSD by the Kabsch algorithm (Kabsch, 1976) based on C_α spatial features of the CDR residues.

Results The LSTM baselines do not involve structure prediction, so we only report the RMSD for the graph-based method. Table 1 reports the performance of AbODE on uncontrolled generation and antigen-conditioned generation tasks, where AbODE outperforms all the baselines on both metrics. AbODE can improve over the SOTA by directly combining the antibody context with the information about the antigen via the attention network, thereby demonstrating the benefits of joint modeling. We also evaluate the biological functionality of the generated antibodies, shown in Fig. 3. Specifically, we considered the following properties:

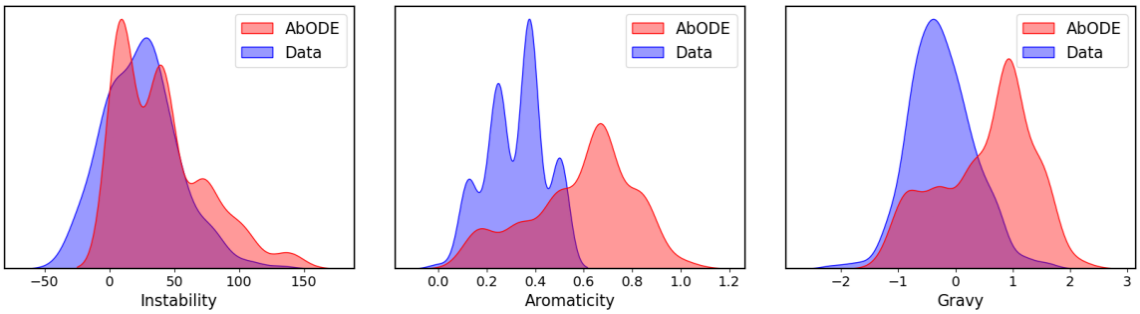

Figure 3: Functional evaluation of generated antibodies vs. data for CDR-H1 unconditional antibody sequence and structure design

- **Gravy**: The Gravy value is calculated by adding the hydropathy value for each residue and dividing it by the length of the sequence (Kyte & Doolittle, 1982)

- **Instability**: The Instability index is calculated using the approach of Guruprasad et al. (1990), which predicts regional instability of dipeptides that occur more frequently in unstable proteins when compared to stable proteins.

- **Aromaticity**: It calculates the aromaticity value of a protein according to Lobry & Gautier (1994). It is simply the relative frequency of Phe+Trp+Tyr.

As our plots demonstrate, AbODE can essentially replicate the behavior of the data in terms of instability and gravy. However, there is some discrepancy in terms of spread concerning aromaticity.

To further evaluate our model, we designed CDR- H3 that binds to a given antigen. We used AAR and RMSD as our scoring metrics. We included RosettaAD (Adolf-Bryfogle et al., 2018), a conventional physics-based baseline for comparison. We benchmark our method on 60 diverse complexes selected by (Adolf-Bryfogle et al., 2018). Note, however, that the training is still conducted on the SAbDab dataset as described in the previous section, where we eliminate the antibodies that overlap with those in RAbD to avoid any data leakage.

The performance of `AbODE` , and its comparison with the baselines, is reported in Table 2. `AbODE` can improve upon the best-performing baseline MEAN while significantly outperforming all the other baselines in terms of both the AAR and the RMSD. In particular, the higher Amino acid recovery rate (AAR) of `AbODE` relative to the other methods demonstrates the ability of the proposed method to learn the underlying distribution of residuals for sequence design.

Table 2: Results on RAbD benchmark. We report Amino acid recovery (AAR) and RMSD for CDR-H3 design. Baselines are from Kong et al. (2023).

Method	AAR % (↑)	RMSD (↓)
RosettaAD	22.50	5.52
LSTM	22.36	(N/A)
C-LSTM	22.18	(N/A)
RefineGNN	29.79	7.55
C-RefineGNN	28.90	7.21
MEAN	36.77	1.81
`AbODE`	**39.95** ± 1.3	**1.54** ± 0.24

Acknowledgements

The calculations were performed using resources made available by the Aalto University Science-IT project. This work has been supported by the Academy of Finland under the *HEALED* project (grant 13342077).

4. Conclusion

We introduced a new generative model `AbODE`, which models the antibody-antigen complex as a joint graph, and performs information propagation using a graph PDE that reduces to a system of coupled residue-specific ODEs. `AbODE` can accurately co-model the sequence and structure of the antigen-antibody complex. In particular, the model can generate a binding antibody sequence and structure with state-of-the-art accuracy for a given antigen.

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., 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.

Alley, E. C., Khimulya, G., Biswas, S., AlQuraishi, M., and Church, G. M. Unified rational protein engineering with sequence-based deep representation learning. *Nature methods*, 16(12):1315–1322, 2019.

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.

Fischman, S. and Ofran, Y. Computational design of antibodies. *Current opinion in structural biology*, 51:156–162, 2018.

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.

Guruprasad, K., Reddy, B. B., and Pandit, M. W. Correlation between stability of a protein and its dipeptide composition: a novel approach for predicting in vivo stability of a protein from its primary sequence. *Protein Engineering, Design and Selection*, 4(2):155–161, 1990.

Ingraham, J., Garg, V., Barzilay, R., and Jaakkola, T. Generative models for graph-based protein design. *Advances in neural information processing systems*, 32, 2019.

Jin, W., Barzilay, D., and Jaakkola, T. Antibody-antigen docking and design via hierarchical structure refinement. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 10217–10227. PMLR, 17–23 Jul 2022a. URL https://proceedings.mlr.press/v162/jin22a.html.

Jin, W., Wohlwend, J., Barzilay, R., and Jaakkola, T. Iterative refinement graph neural network for antibody sequence-structure co-design, 2022b.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., 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.

Kong, X., Huang, W., and Liu, Y. Conditional antibody design as 3d equivariant graph translation, 2023.

Kuroda, D., Shirai, H., Jacobson, M. P., and Nakamura, H. Computer-aided antibody design. *Protein Engineering, Design and Selection*, 25(10):507–522, 06 2012. ISSN 1741-0126. doi: 10.1093/protein/gzs024. URL https://doi.org/10.1093/protein/gzs024.

Kyte, J. and Doolittle, R. F. A simple method for displaying the hydropathic character of a protein. *Journal of molecular biology*, 157(1):105–132, 1982.

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.

Li, Y., Gu, C., Dullien, T., Vinyals, O., and Kohli, P. Graph matching networks for learning the similarity of graph structured objects. In *International conference on machine learning*, pp. 3835–3845. PMLR, 2019.

Lobry, J. and Gautier, C. Hydrophobicity, expressivity and aromaticity are the major trends of amino-acid usage in 999 escherichia coli chromosome-encoded genes. *Nucleic acids research*, 22(15):3174–3180, 1994.

Pantazes, R. and Maranas, C. D. Optcdr: a general computational method for the design of antibody complementarity determining regions for targeted epitope binding. *Protein Engineering, Design & Selection*, 23(11):849–858, 2010.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Pinto, D., Park, Y.-J., Beltramello, M., Walls, A. C., Tortorici, M. A., Bianchi, S., Jaconi, S., Culap, K., Zatta, F., De Marco, A., et al. Cross-neutralization of sars-cov-2 by a human monoclonal sars-cov antibody. *Nature*, 583 (7815):290–295, 2020.

Satorras, V. G., Hoogeboom, E., and Welling, M. E(n) equivariant graph neural networks, 2021. URL https://arxiv.org/abs/2102.09844.

Shi, Y., Huang, Z., Feng, S., Zhong, H., Wang, W., and Sun, Y. Masked label prediction: Unified message passing

model for semi-supervised classification. *arXiv preprint arXiv:2009.03509*, 2020.

Shin, J.-E., Riesselman, A. J., Kollasch, A. W., McMahon, C., Simon, E., Sander, C., Manglik, A., Kruse, A. C., and Marks, D. S. Protein design and variant prediction using autoregressive generative models. *Nature communications*, 12(1):2403, 2021a.

Shin, J.-E., Riesselman, A. J., Kollasch, A. W., McMahon, C., Simon, E., Sander, C., Manglik, A., Kruse, A. C., and Marks, D. S. Protein design and variant prediction using autoregressive generative models. *Nature communications*, 12(1):2403, 2021b.

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.

Verma, Y., Kaski, S., Heinonen, M., and Garg, V. Modular flows: Differential molecular generation. *arXiv preprint arXiv:2210.06032*, 2022.

A. Appendix

A.1. Orientation Matrix

Orientation matrix(Ingraham et al., 2019) defines invariant and locally informative features, using a local coordinate system at each residue i, in terms of the backbone geometry. It is formally defined as,

$$\mathcal{O}_i = [\mathbf{u}_i, \mathbf{n}_i, \mathbf{b}_i \times \mathbf{n}_i] \tag{17}$$

$$\mathbf{u}_i = \frac{\mathbf{x}_i - \mathbf{x}_{i-1}}{||\mathbf{x}_i - \mathbf{x}_{i-1}||},\ \mathbf{b}_i = \frac{\mathbf{u}_i - \mathbf{u}_{i+1}}{||\mathbf{u}_i - \mathbf{u}_{i+1}||},\ \mathbf{n}_i = \frac{\mathbf{u}_i \times \mathbf{u}_{i+1}}{||\mathbf{u}_i \times \mathbf{u}_{i+1}||} \tag{18}$$

where $\mathbf{b}_i$ acts as a negative angle bisector between the vectors $\mathbf{x}_{i-1} - \mathbf{x}_i$ and $\mathbf{x}_{i+1} - \mathbf{x}_i$ and $\mathbf{n}_i$ is the unit normal vector of that plane.

A.2. Connection to Independent E(3)-Equivariant Graph Matching Networks (IEGMNs)

Independent E(3)-Equivariant Graph Matching Networks (Ganea et al., 2021) combine Graph Matching Networks (GMN) (Li et al., 2019) and E(3)-Equivariant Graph Neural Networks (Satorras et al., 2021), to characterize interactions between an input pair of graphs $G_1 = (V_1, E_1), G_2 = (V_2, E_2)$. IEGMNs utilize inter and intra-message passing to update the node features and the spatial encodings. We adopt the notation from (Ganea et al., 2021): m_{ij} denotes the messages between nodes i and j, m_n represents the averaged message over all the neighbors, μ_{ij} represents the intra-connection edge features, and a_{ij} are the attention coefficients. These features create an aggregated external message in μ_1 and μ_2. The aggregated external messages are then used to update the node feature embedding $\mathbf{h}_n$, and the spatial embedding $\mathbf{x}_n$ for all nodes in both graphs.

As outlined in (Table 3), `AbODE` shares strong similarities with IEGMN. Interestingly, both methods compute two kinds of messages (one kind pertains to messages for nodes of the same type/graph, and the other for a different type/graph). The role of μ_{ij} is seem to be played by $\mathbf{m}_{ij}^{\prime,int}$ and $\mathbf{m}_{ij}^{\prime,ext}$ to update the corresponding node and spatial embeddings.

Table 3: `AbODE` as a variant of Independent E(3)-Equivariant Graph Matching Network (IEGMN) applied to interactions among two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. Here, $e_{ij} \in E_1 \cup E_2$; $n \in V_1 \cup V_2$; $\texttt{RBF}(\mathbf{x}_i, \mathbf{x}_j; \sigma) = \texttt{exp}(-||\mathbf{x}_i^{(l)} - \mathbf{x}_j^{(l)}||^2/\sigma)$; h_n and x_n denote, respectively, the node embedding and the spatial embedding; a_{ij} are attention based coefficients; ϕ^x is a real-valued (scalar) parametric function; $\phi^{h,e}$ are parametric functions (MLPs); $\mathbf{f}_{ij}, \mathbf{f}_i$ are the original edge and node features; β, η are scaling parameters and $\mathbf{W}$ is a learnable matrix. For `AbODE` , $\alpha_{i,j}$ are the attention coefficients; $\mathbf{W}_1, \ldots, \mathbf{W}_6$ are learnable weight parameters; d is the hidden size of each head; $\mathcal{N}_{int}(i)$ are the neighbours j of node i such that $k_{ij} = 1$, and $\mathcal{N}_{ext}(i)$ are the neighbours such that $k_{ij} = 2$.

Method	IEGMN layer	`AbODE`
Edge	$\mathbf{m}_{ij} = \varphi^e\left(\mathbf{h}_i^{(l)}, \mathbf{h}_j^{(l)}, \text{RBF}\left(\mathbf{x}_i^{(l)}, \mathbf{x}_j^{(l)}; \sigma\right), \mathbf{f}_{ij}\right)$ $\mathrm{m}_n = \frac{1}{\lvert\mathcal{N}(n)\rvert}\sum_{j\in\mathcal{N}(n)} \mathrm{m}_{nj}$	$\alpha_{i,j} = \text{softmax}\left(\frac{(\mathbf{W}_3\mathbf{z}_i)^\top(\mathbf{W}_4\mathbf{z}_j + \mathbf{W}_6\mathbf{e}_{i,j}).}{\sqrt{d}}\right)$ $m_i' = \sum_{j\in\mathcal{N}_i} \alpha_{i,j}\left(\mathbf{W}_2\mathbf{z}_j + \mathbf{W}_6\mathbf{e}_{ij}\right)$
Intra and Inter connections	$\boldsymbol{\mu}_{ij} = a_{ij}\mathbf{W}\mathbf{h}_j^{(l)}, \forall i \in \mathcal{V}_1, j \in \mathcal{V}_2$ or $i \in \mathcal{V}_2, j \in \mathcal{V}_1$ $\boldsymbol{\mu}_i = \sum_{j\in\mathcal{V}_2} \boldsymbol{\mu}_{ij}, \forall i \in \mathcal{V}_1, \quad \boldsymbol{\mu}_k = \sum_{l\in\mathcal{V}_1} \boldsymbol{\mu}_{kl}, \forall k \in \mathcal{V}_2$	$m_{ij}^{\prime,ext} = \alpha_{i,j}\left(\mathbf{W}_2\mathbf{z}_j + \mathbf{W}_6\mathbf{e}_{ij}\right),\ m_{ij}^{\prime,int} = \alpha_{i,j}\left(\mathbf{W}_2\mathbf{z}_j + \mathbf{W}_6\mathbf{e}_{ij}\right)$ $\mathbf{m}_i^{\prime,int} = \sum_j^{\mathcal{N}_{int}(i)} m_{ij}^{\prime,int},\ \mathbf{m}_i^{\prime,ext} = \sum_j^{\mathcal{N}_{ext}(i)} m_{ij}^{\prime,ext}$
Node embedding	$\mathbf{h}_n^{(l+1)} = (1-\beta)\cdot\mathbf{h}_n^{(l)} + \beta\cdot\varphi^h\left(\mathbf{h}_n^{(l)}, \mathbf{m}_n, \boldsymbol{\mu}_n, \mathbf{f}_n\right)$	$\mathrm{a}_i' = \mathbf{W}_1\mathbf{a}_i + \mathbf{m}_i^{\prime,\text{int}} + \mathbf{m}_i^{\prime,\text{ext}}$
Coordinate embedding	$\mathbf{x}_n^{(l+1)} = \eta\mathbf{x}_n^{(0)} + (1-\eta)\mathbf{x}_n^{(l)} + \sum_{j\in\mathcal{N}(n)}\left(\mathbf{x}_n^{(l)} - \mathbf{x}_j^{(l)}\right)\varphi^x\left(\mathbf{m}_{nj}\right)$	$\mathrm{s}_i' = \mathbf{W}_1\mathbf{s}_i + \mathbf{m}_i^{\prime,\text{int}} + \mathbf{m}_i^{\prime,\text{ext}}$

A.3. Implementation

We implemented AbODE in PyTorch (Paszke et al., 2019). We used three layers of Transformer Convolutional Network (Shi et al., 2020) with hidden embedding dimensions of $128 - 256 - 64$. The ODE solver operated over time-steps $t \in [0, 200]$, where we took the last time step value as the final prediction of the model. The ODE system is solved with the Adaptive heun solver with an adaptive step size. We train the models for 5000 epochs with the Adam optimizer and use a batch size of 300.