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.
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 $V = (V_{Ab}, V_{Ag})$, coordinates $X = (X_{Ab}, X_{Ag})$ and edges $E = (E_{Ab}, E_{Ab-Ag})$ within the antibody as well as between the antibody and the antigen. Each vertex $v \in \mathcal{A}\{Arg, His, \ldots\}$ is one of 20 amino acids. We treat the labels with a Categorical distribution, such that the label features $a_i \in \mathbb{R}^{20}$ 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 $x_i$ we compute its spatial features $s_i = (r_i, \alpha_i, \gamma_i)$, where

$$r_i = \|u_i\|, \quad u_i = x_{i+1} - x_i$$

$$\alpha_i = \cos^{-1}\left(\frac{\langle u_i, u_{i-1} \rangle}{\|u_i\| \cdot \|u_{i-1}\|}\right)$$

$$\gamma_i = \cos^{-1}\left(\frac{\langle u_i, n_i \rangle}{\|u_i\| \cdot \|n_i\|}\right), \quad n_i = u_i \times u_{i-1}.$$

Here, $r_i$ is the distance between consecutive residues $x_i$ and $x_{i+1}$, $\alpha_i$ is the co-angle of residue $i$ w.r.t previous and next residue, $\gamma_i$ is the azimuthal angle of $i$’s local plane, and $n_i$ is the normal vector. The full residue state $x_i = [a_i, s_i]$ concatenates the label features $a_i$ and the spatial features $s_i$.

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 $V = (V_{Ab}, V_{Ag})$, coordinates $X = (X_{Ab}, X_{Ag})$ and edges $E = (E_{Ab}, E_{Ab-Ag})$ within the antibody as well as between the antibody and the antigen. Each vertex $v \in \mathcal{A}\{Arg, His, \ldots\}$ is one of 20 amino acids. We treat the labels with a Categorical distribution, such that the label features $a_i \in \mathbb{R}^{20}$ 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 $x_i$ we compute its spatial features $s_i = (r_i, \alpha_i, \gamma_i)$, where

$$r_i = \|u_i\|, \quad u_i = x_{i+1} - x_i$$

$$\alpha_i = \cos^{-1}\left(\frac{\langle u_i, u_{i-1} \rangle}{\|u_i\| \cdot \|u_{i-1}\|}\right)$$

$$\gamma_i = \cos^{-1}\left(\frac{\langle u_i, n_i \rangle}{\|u_i\| \cdot \|n_i\|}\right), \quad n_i = u_i \times u_{i-1}.$$

Here, $r_i$ is the distance between consecutive residues $x_i$ and $x_{i+1}$, $\alpha_i$ is the co-angle of residue $i$ w.r.t previous and next residue, $\gamma_i$ is the azimuthal angle of $i$’s local plane, and $n_i$ is the normal vector. The full residue state $x_i = [a_i, s_i]$ concatenates the label features $a_i$ and the spatial features $s_i$.

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 antibody-antigen complex as a 3D graph $G = (V, E, X)$, with antibody $Ab$ and antigen $Ag$ amino-acid residues as vertices $V = (V_{Ab}, V_{Ag})$, coordinates $X = (X_{Ab}, X_{Ag})$ and edges $E = (E_{Ab}, E_{Ab-Ag})$ within the antibody as well as between the antibody and the antigen. Each vertex $v \in \mathcal{A}\{Arg, His, \ldots\}$ is one of 20 amino acids. We treat the labels with a Categorical distribution, such that the label features $a_i \in \mathbb{R}^{20}$ 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 $x_i$ we compute its spatial features $s_i = (r_i, \alpha_i, \gamma_i)$, where

$$r_i = \|u_i\|, \quad u_i = x_{i+1} - x_i$$

$$\alpha_i = \cos^{-1}\left(\frac{\langle u_i, u_{i-1} \rangle}{\|u_i\| \cdot \|u_{i-1}\|}\right)$$

$$\gamma_i = \cos^{-1}\left(\frac{\langle u_i, n_i \rangle}{\|u_i\| \cdot \|n_i\|}\right), \quad n_i = u_i \times u_{i-1}.$$

Here, $r_i$ is the distance between consecutive residues $x_i$ and $x_{i+1}$, $\alpha_i$ is the co-angle of residue $i$ w.r.t previous and next residue, $\gamma_i$ is the azimuthal angle of $i$’s local plane, and $n_i$ is the normal vector. The full residue state $x_i = [a_i, s_i]$ concatenates the label features $a_i$ and the spatial features $s_i$.

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

$$e_{ij} = (\Delta z_{ij}, i - j, \text{RBF} \left(\|s_i - s_j\|\right)), \quad \Delta z_{ij} = \{\Delta a_{ij}, \Delta s_{ij}\},$$

$$O_i^\top (s_i, s_i^\alpha, s_j^\alpha), \quad O_j^\top (s_j, s_j^\alpha), \quad k_{ij}.$$

These include state differences $\Delta z_{ij} = \{\Delta a_{ij}, \Delta s_{ij}\}$, backbone distance $i - j$, and spatial distance $\text{RBF}(\|s_i, s_j\|)$ (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 $O_i$ (Ingraham et al., 2019), and the $O_i^\top O_j$ 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 \( z(t) \) over time \( t \in \mathbb{R}_+ \). We initialize the initial state \( z(0) \) 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 (Kong et al., 2023). Co-ordinates are initialized with the even distribution between the residue right before CDRs and the one right after CDRs (Jin et al., 2022b; Kong et al., 2023).

The sequence loss is quantified in terms of the cross-entropy between the true label \( \alpha_{ni}^{true} \) and the label distribution \( \alpha_{ni} \) predicted by the model, i.e.,

\[
L_{seq} = \frac{1}{N} \sum_{n=1}^{N} \sum_{i=1}^{M_i} CE \left( \alpha_{ni}^{true}, \alpha_{ni} \right) \tag{13}
\]

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:

\[
L_{structure} = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{M} \sum_{i=1}^{M_i} \lambda \left( L_{angle}^{ni} + L_{radius}^{ni} \right) \tag{14}
\]

For each residue angle pair \((\alpha, \gamma)\) we compute the negative log of the von-Mises likelihood

\[
L_{angle}^{ni} = \sum_{k} \sum_{\theta \in \{\alpha, \gamma\}} \log \mathcal{M} \left( \theta_{ik}^n | \theta_{ik}^{true}, \kappa \right) \tag{15}
\]
We transform the label features $\sigma$ where $\kappa$ 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.

$$L_{\text{radius}}^{ni} = \sum_k \log \mathcal{N}(r_{ik}^{n}, r_{ik}^{\text{true}}, \sigma_r^2)$$

(16)

where $\sigma_r^2$ is the radius variance. Here $\lambda$ is the polar loss weight, set to $\lambda = 0.8$. We set $\kappa = 10$, $\sigma_r^2 = 0.1$ 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 $z(T) = [\mathbf{a}(T), \mathbf{s}(T)]$. 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 unconditional 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 $C_{\alpha}$ 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:

<table>
<thead>
<tr>
<th>Method</th>
<th>CDR-H1</th>
<th>CDR-H2</th>
<th>CDR-H3</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM</td>
<td>6.79</td>
<td>7.21</td>
<td>9.70</td>
</tr>
<tr>
<td>AR-GNN</td>
<td>6.44</td>
<td>2.97</td>
<td>3.63</td>
</tr>
<tr>
<td>RefineGNN</td>
<td>6.09</td>
<td>1.18</td>
<td>8.38</td>
</tr>
<tr>
<td>AbODE</td>
<td>4.25</td>
<td>0.73</td>
<td>2.01</td>
</tr>
</tbody>
</table>

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).
AbODE: Ab initio antibody design using conjoined ODEs

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).

<table>
<thead>
<tr>
<th>Method</th>
<th>AAR (%) (↑)</th>
<th>RMSD (↓)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RosettaAD</td>
<td>22.50</td>
<td>5.52</td>
</tr>
<tr>
<td>LSTM</td>
<td>22.36</td>
<td>(N/A)</td>
</tr>
<tr>
<td>C-LSTM</td>
<td>22.18</td>
<td>(N/A)</td>
</tr>
<tr>
<td>RefineGNN</td>
<td>29.79</td>
<td>7.55</td>
</tr>
<tr>
<td>C-RefineGNN</td>
<td>28.90</td>
<td>7.21</td>
</tr>
<tr>
<td>MEAN</td>
<td>36.77</td>
<td>1.81</td>
</tr>
<tr>
<td>AbODE</td>
<td><strong>39.95 ± 1.3</strong></td>
<td><strong>1.54 ± 0.24</strong></td>
</tr>
</tbody>
</table>

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

AbODE: Ab initio antibody design using conjoined ODEs


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


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,

$$ O_i = [u_i, n_i, b_i \times n_i] $$

$$ u_i = \frac{x_i - x_{i-1}}{||x_i - x_{i-1}||}, \quad b_i = \frac{u_i - u_{i+1}}{||u_i - u_{i+1}||}, \quad n_i = \frac{u_i \times u_{i+1}}{||u_i \times u_{i+1}||} $$

where $b_i$ acts as a negative angle bisector between the vectors $x_{i-1} - x_i$ and $x_{i+1} - x_i$ and $n_i$ 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, $\mu_{ij}$ represents the intra-connection edge features, and $a_{ij}$ are the attention coefficients. These features create an aggregated external message in $\mu_1$ and $\mu_2$. The aggregated external messages are then used to update the node feature embedding $h_i$, and the spatial embedding $x_i$, 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 $\mu_{ij}$ is seen to be played by $m_{ij}^{int}$ and $m_{ij}^{ext}$ 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, $e_{ij} \in E_1 \cup E_2$; $n \in V_1 \cup V_2$; $\text{RBF}(x_i, x_j; \sigma) = \exp(-||x_i^{(l)} - x_j^{(l)}||^2/\sigma)$; $h_n$ and $x_n$ denote, respectively, the node embedding and the spatial embedding; $a_{ij}$ are attention based coefficients; $\phi^x$ is a real-valued (scalar) parametric function; $\phi^{h,e}$ are parametric functions (MLPs); $f_{ij}, f_i$ are the original edge and node features; $\beta, \gamma$ are scaling parameters and $W$ is a learnable matrix. For AbODE, $\alpha_{ij}$, $\alpha_{ij}$ are the attention coefficients; $W_1, \ldots, W_6$ are learnable weight parameters; $d$ is the hidden size of each head and $N_{\text{int}}(i)$ are the neighbours $j$ of node $i$ such that $k_{ij} = 1$, and $N_{\text{ext}}(i)$ are the neighbours such that $k_{ij} = 2$.

<table>
<thead>
<tr>
<th>Method</th>
<th>IEGMN layer</th>
<th>AbODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge</td>
<td>$m_{ij} = \phi^x(h_i, h_j, \text{RBF}(x_i, x_j; \sigma), f_{ij})$</td>
<td>$\alpha_{ij} = \text{softmax}(\frac{W_1 e_i + W_2 e_j}{W_3 e_i + W_4 e_j})$</td>
</tr>
<tr>
<td>Intra and Inter connections</td>
<td>$\mu_{ij} = a_{ij} W h_i^{(l)}, \forall i \in V_1, j \in V_2$ or $i \in V_2, j \in V_1$</td>
<td>$m_{ij}^{int} = \alpha_{ij}(W_5 z_i + W_6 e_{ij}); m_{ij}^{int} = \alpha_{ij}(W_7 z_i + W_8 e_{ij})$</td>
</tr>
<tr>
<td>Node embedding</td>
<td>$h_i^{(l)} = (1 - \beta) h_i^{(l-1)} + \beta \cdot \phi^x(h_i^{(l-1)}, m_n, \mu_n, f_i)$</td>
<td>$m_i^{int} = \sum_{j \in N_{\text{int}}(i)} m_{ij}^{int}; m_i^{ext} = \sum_{j \in N_{\text{ext}}(i)} m_{ij}^{ext}$</td>
</tr>
<tr>
<td>Coordinate embedding</td>
<td>$x_i^{(l+1)} = \eta x_i^{(l)} + (1 - \eta)(x_i^{(l)} - \sum_{j \in N_{\text{int}}(i)} (x_j^{(l)} - x_i^{(l)})) \phi^x(m_n)$</td>
<td>$s_i = W_9 s_i + m_i^{int} + m_i^{ext}$</td>
</tr>
</tbody>
</table>

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.