MODELING STATIC AND DYNAMIC PROTEIN STRUCTURE FROM 2D INFRARED SPECTRUM WITH STOCHASTIC INTERPOLANT

Anonymous authors
Paper under double-blind review

ABSTRACT

A protein's function is intrinsically linked to its dynamic structure, yet predicting these conformational changes in real-time remains a central challenge in molecular biology. While two-dimensional infrared (2DIR) spectroscopy provides a powerful experimental window into these dynamics, translating its complex, low-dimensional signals into high-resolution 3D structures is a formidable interpretive hurdle. Current machine learning approaches typically sidestep this challenge by predicting auxiliary information for other computationally expensive tools, creating a slow and indirect workflow. In this paper, we introduce a direct, end-to-end solution: an equivariant generative pipeline based on stochastic interpolants augmented with a length prediction head. The model pretrained on a dataset of static protein bypasses intermediate steps, learning to generate 3D protein structures directly from their corresponding 2DIR spectra, outperforming previous methods. We further show that the model can be finetuned on correlated samples drawn from simulated reversible folding trajectories to improve its predictive power for dynamic protein structure prediction.

1 INTRODUCTION

Understanding protein function requires insights into the dynamic evolution of their atomic structures , which has led to significant efforts in developing tools for structure determination . Advances in machine learning have revolutionized the prediction of a protein's fully folded three-dimensional structure from its primary amino acid sequence, with models like AlphaFold (Jumper et al., 2021) and RoseTTAFold (Baek et al., 2021; Krishna et al., 2024) significantly enhancing our understanding of static protein structures. However, the intermediate states that control proteins' dynamic behavior are less explored. This hinders our comprehension of critical processes such as transmembrane transport, ligand binding, conformational changes, and protein folding. Real-time monitoring of protein structures and dynamics is therefore essential for a complete understanding of their function.

Spectroscopic techniques have long been used to monitor protein dynamics, as they offer detailed temporal and spatial insights into structural changes. Among these, two-dimensional infrared (2DIR) spectroscopy stands out for its high spectral resolution and ability to capture nanometer-scale conformational changes on picosecond to nanosecond timescales Tumbic et al. (2021); Hochstrasser (2008); Ye et al. (2025); Mukherjee et al. (2012). Compared to one-dimensional infrared spectroscopy, which produces congested spectra for larger molecules, 2DIR utilizes an additional frequency dimension to achieve higher resolution. This added clarity allows for the detection of interactions between vibrational modes, making 2DIR especially effective for analyzing complex systems and accounting for environmental effects. Furthermore, 2DIR data can be generated through both theoretical calculations and experimental measurements, bridging the gap between theory and experiment.

Despite its advantages, translating the complex, low-dimensional signals from 2DIR spectroscopy into high-resolution 3D structures remains a formidable interpretive hurdle . While cross-peaks in 2DIR spectra provide valuable information about atomic distances, these data alone do not provide a complete picture of the overall protein structure. This presents a challenge analogous to that

in NMR, where resolving a full structure requires hundreds of experimentally derived constraints. Extracting clear structural insights from the rich map of vibrational couplings in 2DIR spectra has thus necessitated the use of computational methods.

However, current machine learning approaches typically sidestep the direct interpretation challenge. They often serve as intermediate steps, predicting auxiliary information or spectral descriptors for other computationally expensive tools, which creates a slow and indirect workflow. In this paper, we introduce a direct, end-to-end solution to this problem: an equivariant generative pipeline based on stochastic interpolants. Our model bypasses intermediate steps, learning to generate 3D protein backbone structures directly and efficiently from their corresponding 2DIR spectra (Ye et al., 2025). This allows the model to capture more faithfully the probabilistic nature of the procedure and preliminarily outperforms prior approach.

2 Stochastic Interpolants and Probability Flow ODE

We first provide a recap of the stochastic interpolant framework for generative modeling. The **stochastic interpolant** framework offers a unified perspective for constructing the probability flow ODEs central to many modern generative models Albergo & Vanden-Eijnden (2023). This approach defines a time-continuous process, or interpolant, that connects samples from a source distribution ρ_0 (e.g., a Gaussian) and a target data distribution ρ_1. The interpolant takes the form:

$$I_t = \alpha_t x_0 + \beta_t x_1, \tag{1}$$

where $x_0 \sim \rho_0$, $x_1 \sim \rho_1$, and (α_t, β_t) are scalar scheduling functions defining the path for $t \in [0, 1]$.

This stochastic process has the same law of a deterministic probability flow $\dot{x}_t = b_t(x_t)$, where the velocity field is given by:

$$b_t(x_t) = \mathbb{E}[\dot{I}_t \mid I_t = x_t] \tag{2}$$

where this expectation is taken over Law(I_t)

A key advantage of this formulation is that $b_t(x)$ can be learned directly by training a neural network $\hat{b}$ with a simple mean-squared error objective:

$$\min_{\hat{b}} \int_0^1 \mathbb{E}_{x_0, x_1}[\|\hat{b}_t(I_t) - \dot{I}_t\|^2] dt. \tag{3}$$

For generation, one samples a prior point $x_0 \sim \rho_0$ and uses a numerical solver to integrate the learned ODE $\dot{x}_t = \hat{b}_t(x_t)$ from $t = 0$ to $t = 1$, yielding a final sample $x_1 \sim \rho_1$. The framework's generality allows it to recover various generative models; for instance, linear schedules correspond to rectified flows, while other choices connect to diffusion models. We provide a more extensive mathematical exposition in Appendix 8.

3 Method for Pretraining

3.1 Static Protein as Pretraining Data

Following prior work (Ye et al., 2025), the foundational dataset was compiled from the **RCSB Protein Data Bank** and **SWISS-PROT** library, consisting of **49,547 protein structures**, with each protein limited to a maximum of 100 residues.

Due to the scarcity of experimental 2DIR spectral data, a theoretical approach was employed to create the dataset for model training. **Input Features (2DIR Spectra):** For each of the 49,547 protein conformations, two-dimensional infrared (2DIR) spectra were theoretically simulated. This was achieved by simulating **Frenkel exciton Hamiltonian** focused on the amide I spectral window ($1,575$ cm^{-1} to $1,725$ cm^{-1}). These simulated 2DIR signals were then converted into $3 \times 224 \times 224$ **RGB images** to serve as the input for the neural network. One notable difference between this task and prior problems is that during generation the length of the protein structure is not known priori, this implies that model has to learn to predict the length on the fly, as such we include the length of each protein structure in the dataset.

Figure 1: Two samples of simulated 2DIR used as input to a conditional generative models. Similar samples

3.2 Modeling and Training

As the length of a protein is unknown given only the 2D IR image, we separately employ a length prediction head. We note that this is not theoretically grounded as this assumes the distribution of x_1 given the conditioning c is concentrated only takes one length, however, we observe this to be empirically effective in practice. To train this length prediction head, we employ a simple regression loss:

$$L_{\text{length}}[g] = \mathbb{E}_{x_1,c}[g(x_1) - \text{len}(x_1)] \tag{4}$$

where $\text{len}(x_1)$ notes the length of x_1 before pading.

For generation, we adopt an interpolant objective with a linear schedule that is $\alpha_t = t$ and $\beta_t = 1-t$ with the base distribution $\rho_1 = N(0, 1)$, while simple, such choice has been proven very effective in scaling most notably for image generation. We do not employ any weighting factor in the objective. As this is a conditional generation task, the vector field further takes in the 2DIR signal as an input, we additionally condition on the length of data, given in the form of a boolean mask. Formally let d be the dimensionality be the padded data, c be the conditioning variable (2DIR signal) defined jointly with the interpolant, we attempt to learn $\hat{b}_t$ through the loss functional:

$$L[\hat{b}_t] = \int_0^1 \mathbb{E}_{x_1,c,x_0}[||\hat{b}_t(I_t, c, \text{len}(x_1)) - \dot{I}_t||] \, dt \tag{5}$$

the notable difference between the previous equaion is that the vector field $\hat{b}_t$ is now learnt in a way that is amortised over all c.

3.3 Architecture

3.3.1 Conditional Encoder

The 2D IR image c is encoded into a fixed-size latent vector z_c, which provides global conditioning information. The encoder is a lightweight CNN consisting of two strided convolutional layers with ReLU activations, which downsample the image from 128×128 to 32×32. The resulting feature map is flattened and passed through a linear layer to produce a final 512-dimensional embedding.

3.3.2 Equivariant processing

To incorporate geometric inductive biases, this model utilizes Geometric Vector Perceptron (GVP) layers (Jing et al., 2021). The node coordinates x_t (which are grade 1 vector features) are combined with initial scalar features (grade 0 tensors), which are also influenced by the global conditions z_c and z_t. Each node's combined scalar and vector features are then processed through a GVP block. Within this block, the vector features are processed through dedicated operations, and the scalar features are updated via a ϕ^s node-wise MLP. This specialized architecture ensures that the output feature set—consisting of both updated scalar and vector features—strictly preserves E(3) equivariance (for the vectors) and invariance (for the scalars). Hyperperameters are listed in 7.

3.4 SAMPLING

At inference time, the model generates a 3D protein structure directly from a 2DIR spectrum through a two-step procedure: (1) predict the protein's length, and (2) generate the corresponding 3D structure by integrating a learned conditional vector field. The predicted length is used to construct a boolean mask that indicates valid residue positions, which is provided to the vector field during ODE integration.

This section presents the full inference pipeline in pseudocode form.

Algorithm 1 Structure Generation from 2DIR Spectrum using Euler ODE Solver

Require: 2DIR spectrum c, trained length predictor g, trained vector field $\hat{b}_t$, maximum length d, step size h

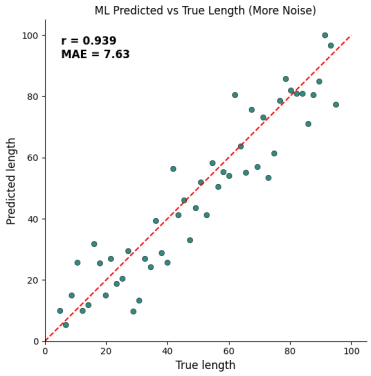
$\hat{\ell} \leftarrow \text{round}(g(c))$ # Predict protein length
$m \leftarrow \text{Mask}(\hat{\ell}, d)$ # Binary mask of length $\hat{\ell}$
$x_0 \sim \mathcal{N}(0, I) \in \mathbb{R}^{d \times 3}$ # Sample Gaussian noise
Initialize $x \leftarrow x_0$
for t **in** 0 **to** $1 - h$ **step** h **do**
  $v \leftarrow \hat{b}_t(x, c, m)$ # Compute vector field at time t
  $x \leftarrow x + h \cdot v$ # Euler update
end for
return $x[m]$ # Return only valid residues

This procedure produces a variable-length 3D backbone structure conditioned only on t.

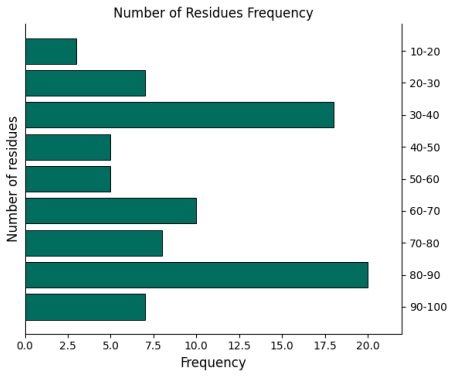
4 EVALUATING PRETRAINED MODEL

4.1 ON THE PERFORMANCE OF LENGTH PREDICTION

To evaluate the performance of our module we first evaluate on the ability of the model to accurately recover the length of the protein with the length prediction head. From Figure 2a, it can be seen that the length predicted by the model is strongly correlated with the length distribution of the data with a $R^2 = 0.939$. Whilst we used different modeling technique, this aligns with experimental results reported by (Ye et al., 2025)



(a) Predicted length from the trained model against the length distribution in the data for a evenly sampled subset of residue in the test dataset.



(b) The length distribution of the test dataset. It can be seen that the length distribution has two distinct modes. Yet, the model seems to perform equally well across different lengths.

Figure 2: Length prediction and data distribution plots.

4.2 On the quality static protein generation

We now investigate the quality of protein structure generated by the model on static generation. To measure the quality, we report the root mean square deviation of the generated samples measured in terms of angstrom. Notably, to isolate the quality of the model from the length prediction module, we supply the model with the number of atoms when running the prediction.

We compare our model to the method in Ye et al. (2025) as a baseline, which operates by predicting a static hamiltonian and using it for gradient descent. We further compare to a non-equivariant model based on the transformer model, where we only encode basic permutation invariance via removal of the positional encoding. To report confidence intervals, we compute the standard deviation over 10 generated samples for our model. We do not report standard deviation for baseline method from Ye et al. (2025) as they employ a deterministic pipeline for inference, as opposed to a generative pipeline, making a fair comparison impossible.

As reported in Figure 3, the interpolant models are able to generate structure of significantly lower RMSD than the baseline methods, showing that the generative nature of the pipeline is able to capture more nuances than the deterministic pipeline in Ye et al. (2025). It can also be seen that the GVP model is able to generate structures that are more aligned with the test data, supporting the fact that the equivariant modeling is useful.

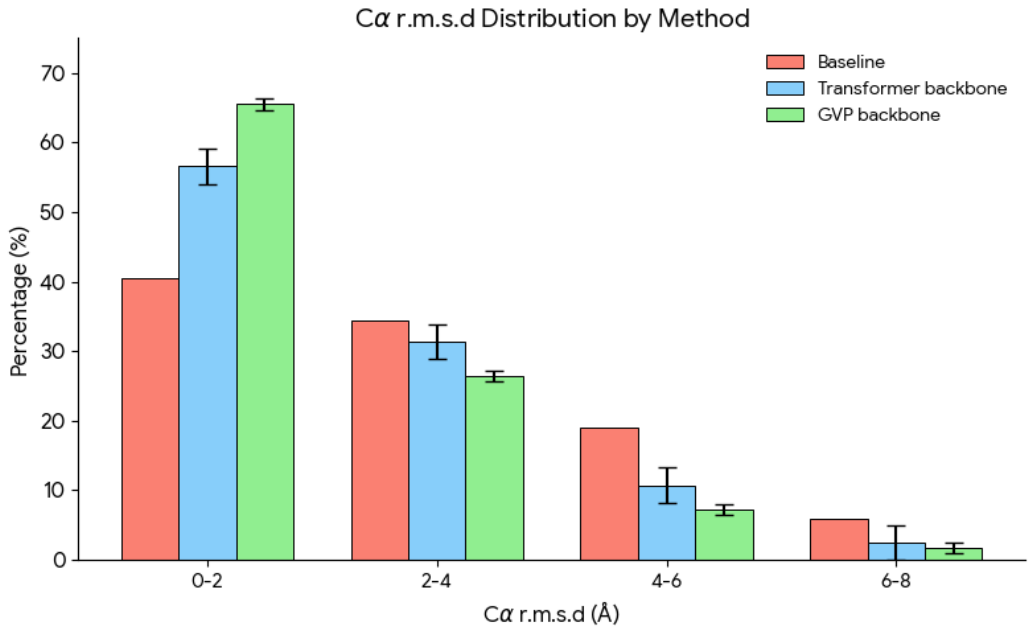


Figure 3: Root Mean Square Deviation between the protein structure generated by our methods and baseline. Notably, the generative models significantly more protein structure has RMSD lower than 2 angstrom.

5 Transfer learning to dynamic protein structures

To assess the model's ability to capture temporal conformational changes, we adopted the evaluation framework described in Ye et al. (2025). We selected ten representative fast-folding proteins (10–80 residues) spanning α, β, and mixed α/β topologies following Ye et al. (2025).

Data Sampling and Spectral Mapping. Ground-truth trajectories were derived from Anton-based MD simulations, capturing reversible folding on microsecond to millisecond scales. We sampled 10,000 conformations per protein at uniform intervals, calculating the corresponding 2DIR spectra to serve as high-dimensional geometric descriptors.

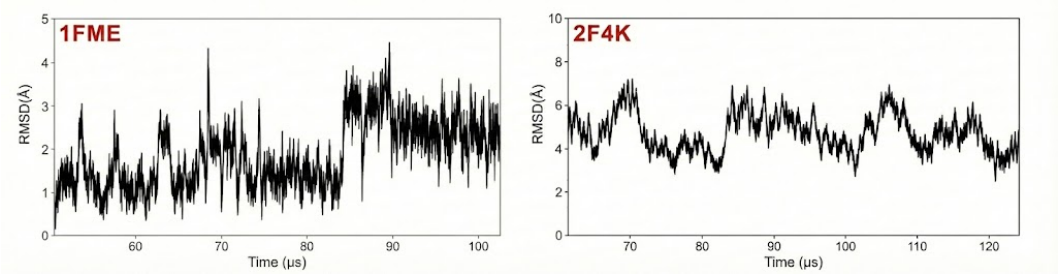


Figure 4: RMSD of generated samples along the simulated trajectories. It can be seen that the RMSD is consistent over the time indicating the model is successfully able to learn how the protein structure involves through time.

Transfer Learning and Partitioning. Following the protocol in Ye et al. (2025), we utilized transfer learning to refine pretrained weights. The dataset was partitioned into a 50% training split for fine-tuning and a 50% hold-out test set. The test partition was specifically chosen to include alternating folding and unfolding events to rigorously evaluate the model's transferability and dynamic tracking. It is worth noting all samples protein is only used in either training or test, as samples from the same simulated dynamics are auto-correlated.

Results. Model performance was quantified by comparing predicted structures against reference MD snapshots. We compute the RMSD across the entire trajectory, and compare with previous work Ye et al. (2025) as a baseline. The results summarised in Table 1 indicate that while the pre-trained model faces a significant distribution shift when applied zero-shot to dynamic MD trajectories, subsequent fine-tuning allows the stochastic interpolant to capture trajectory-specific nuances, ultimately outperforming the deterministic baseline by 13%. In Figure 4, we plot the RMSD between generated samples and the ground truth along simulated trajectories, which is constantly low across time, indicating the model has successfully learnt how the structure changes over time.

Table 1: Comparative performance for dynamic protein structure generation. Values represent the mean backbone RMSD.(Å) across ten test trajectories.

Metric	**Baseline (Finetuned)**	**Ours (Pretrained)**	**Ours (Finetuned)**
Mean Backbone RMSD (↓)	2.51	3.56	2.18

6 Limitation, Conclusion and Future work

We introduced a novel, end-to-end generative pipeline for the direct translation of two-dimensional infrared (2DIR) spectra into high-resolution 3D protein backbone structures. Our model utilizes the stochastic interpolant framework, conditioned on the 2DIR signal architecture to enforce geometric inductive biases.

This approach successfully bypasses the intermediate, computationally expensive steps required by previous methods. This work provides a fast, geometrically-aware tool for interpreting dynamic 2DIR data, representing a critical step toward real-time monitoring of protein conformational changes. One important limitation of our modeling is that our model only works for protein with fewer than 100 residues, as future direction, it is important to consider possibilties to extend the method to protein to more number of residues, this is in contrast to previous method Ye et al. (2025), which support generating beyond the length distribution of the training dataset.

Better evaluation metric should be explored to measure the performance of the pipeline when both the generation and length prediction are used in conjunction. The commonly used RMSD metric can also be misleading for evaluating a generative models, as it fails to capture the degree to which the model can capture the full generation

REFERENCES

Michael Samuel Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=li7qeBbCR1t.

Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Jue Wang, Han Song, and David Baker. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021. ISSN 0036-8075. doi: 10.1126/science.abj8754.

Robin M. Hochstrasser. Amide i two-dimensional infrared spectroscopy of proteins. *Accounts of Chemical Research*, 41(2):239–248, 2008. doi: 10.1021/ar700188n. URL https://pubs.acs.org/doi/10.1021/ar700188n.

Bo Jing, Victor Garcia Satorras, and Max Welling. Geometric vector perceptrons. In *International Conference on Learning Representations*, 2021. URL https://www.google.com/search?q=https://openreview.net/forum%3Fid%3DtBv35f2-9A.

John M. Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A A Kohl, Andy Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with alphafold. *Nature*, 596:583 – 589, 2021. URL https://api.semanticscholar.org/CorpusID:235959867.

Rohith Krishna, Jue Wang, Woody Ahern, Pascal Sturmfels, Preetham Venkatesh, Indrek Kalvet, Gyu Rie Lee, Felix S. Morey-Burrows, Ivan Anishchenko, Ian R. Humphreys, Ryan McHugh, Dionne Vafeados, Xinting Li, George A. Sutherland, Andrew Hitchcock, C. Neil Hunter, Minkyung Baek, Frank DiMaio, and David Baker. Generalized biomolecular modeling and design with rosettafold all-atom. *Science*, 384(6693):343–350, 2024. ISSN 0036-8075. doi: 10.1126/science.adl2528.

Prabuddha Mukherjee, Itamar Kass, Sanjeev K. Satija, Pawel A. Zaleski, Hoi-Sung Chung, and Andrei Tokmakoff. Protein dynamics studied with ultrafast 2d ir vibrational echo spectroscopy. *Accounts of Chemical Research*, 45(11):1957–1967, 2012. doi: 10.1021/ar300057h. URL https://pubs.acs.org/doi/abs/10.1021/ar300057h.

Goran W. Tumbic, Md Yeathad Hossan, and Megan C. Thielges. Protein dynamics by two-dimensional infrared spectroscopy. *Annual Review of Analytical Chemistry*, 14(1):299–321, 2021. doi: 10.1146/annurev-anchem-091520-091009. URL https://www.annualreviews.org/doi/abs/10.1146/annurev-anchem-091520-091009.

Sheng Ye, Lvshuai Zhu, Zhicheng Zhao, Fan Wu, Zhipeng Li, BinBin Wang, Kai Zhong, Changyin Sun, Shaul Mukamel, and Jun Jiang. Ai protocol for retrieving protein dynamic structures from two-dimensional infrared spectra. *Proceedings of the National Academy of Sciences*, 122(7):e2424078122, 2025. doi: 10.1073/pnas.2424078122. URL https://www.pnas.org/doi/abs/10.1073/pnas.2424078122.

7 HYPERPARAMETER TUNING AND SETTINGS

We considered the following hyperparameter settings. The best model was chosen by taking the model with the lowest validation set loss. Hidden Features denote the number of features for each node/token in the network layers. Embedding Size denotes the size of the node/token embeddings in the input. For the GVP-inp model, Layers denotes the number of message passing layers in the GNN architecture.

Table 2: Hyperparameter settings used for the Transformer and GVP models.

Parameters	Search space	Transformer	GVP
Static			
Batch Size		32	32
Sampling steps T		N/A	500
Number of steps		150000	150000
Seed		1	1
Tuned			
Self-Attention	True, False	**True**	N/A
Attention mechanism	True, False	N/A	False
Hidden Features	32, 64, 128, 256	256	32
Embedding Size	32, 64, 128, 256	256	256
Learning rate	5e-3, 2e-3, 1e-3, 5e-4, 1e-4	1e-3	5e-4
Transformer Layers	4, 5, 6, 7	7	N/A
GNN Layers	4, 5, 6, 7	N/A	6

8 A MATHEMATICAL EXPOSITION OF STOCHASTIC INTERPOLANT

In this section, we provide a formal derivation of the velocity field and the corresponding probability flow ODE as established by Albergo & Vanden-Eijnden (2023).

8.1 THE INTERPOLANT PROCESS

Let ρ_0 be the source distribution and ρ_1 be the target distribution. We define the stochastic interpolant I_t as:

$$I_t = \alpha_t x_0 + \beta_t x_1, \quad x_0 \sim \rho_0, x_1 \sim \rho_1 \tag{6}$$

where $t \in [0, 1]$. The functions α_t and β_t are differentiable scalar paths such that $(\alpha_0, \beta_0) = (1, 0)$ and $(\alpha_1, \beta_1) = (0, 1)$.

8.2 THE PROBABILITY FLOW ODE

The marginal probability density $\rho_t(x)$ of the process I_t is given by:

$$\rho_t(x) = \mathbb{E}[\delta(I_t - x)] \tag{7}$$

where δ is the Dirac delta function. Taking the time derivative of the density:

$$\frac{\partial \rho_t(x)}{\partial t} = \mathbb{E}\left[\frac{\partial}{\partial t}\delta(I_t - x)\right] \tag{8}$$

$$= \mathbb{E}\left[\dot{I}_t \cdot \nabla\delta(I_t - x)\right] \tag{9}$$

$$= -\nabla \cdot \mathbb{E}[\dot{I}_t \delta(I_t - x)] \tag{10}$$

To satisfy the continuity equation $\frac{\partial \rho_t}{\partial t} + \nabla \cdot (\rho_t b_t) = 0$, we identify the velocity field $b_t(x)$ as:

$$b_t(x) = \frac{\mathbb{E}[\dot{I}_t \delta(I_t - x)]}{\rho_t(x)} = \mathbb{E}[\dot{I}_t \mid I_t = x] \tag{11}$$

This confirms that the deterministic ODE $\dot{x}_t = b_t(x_t)$ generates the same marginal densities ρ_t as the stochastic process I_t.

8.3 THE TRAINING OBJECTIVE

The velocity field $b_t(x)$ is the L^2-projection of the time derivative $\dot{I}_t$ onto the space of functions of I_t. Given a neural network $\hat{b}_t(x; \theta)$, we minimize the mean-squared error:

$$\mathcal{L}(\theta) = \int_0^1 \mathbb{E}_{x_0, x_1}\left[\|\hat{b}_t(I_t; \theta) - \dot{I}_t\|^2\right] dt \tag{12}$$

Expanding the norm, we observe:

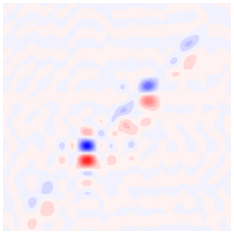
$$\mathbb{E}[\|\hat{b}_t(I_t) - \dot{I}_t\|^2] = \mathbb{E}[\|\hat{b}_t(I_t)\|^2] - 2\mathbb{E}[\hat{b}_t(I_t) \cdot \dot{I}_t] + \mathbb{E}[\|\dot{I}_t\|^2] \tag{13}$$

By the law of total expectation, the cross term becomes:

$$\mathbb{E}[\hat{b}_t(I_t) \cdot \dot{I}_t] = \mathbb{E}[\mathbb{E}[\hat{b}_t(I_t) \cdot \dot{I}_t \mid I_t]] = \mathbb{E}[\hat{b}_t(I_t) \cdot b_t(I_t)] \tag{14}$$

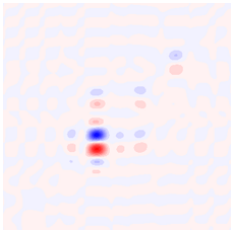
Thus, the objective $\mathcal{L}(\theta)$ is equivalent to minimizing $\mathbb{E}[\|\hat{b}_t(I_t) - b_t(I_t)\|^2]$ up to a constant term, ensuring that the learned network converges to the true conditional expectation $b_t(x)$.

9 SAMPLES OF GENERATED DATA

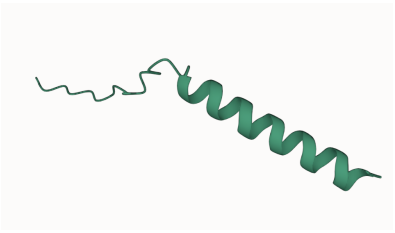


(a) Infrared spectrum (Sample 1)

(b) Protein structure (Sample 1)



(c) Infrared spectrum (Sample 2)



(d) Protein structure (Sample 2)

Figure 5: Samples of infrared spectra from the dataset and their corresponding generated protein structures.