Capturing Protein Dynamics: Encoding Temporal and Spatial Dynamics from Molecular Dynamics Simulations

Vignesh C. Bhethanabotla
Chemistry and Chemical Engineering
California Institute of Technology
vbhethan@caltech.edu

Mohammadamin Tavakoli
Computing and Mathematical Sciences
California Institute of Technology
amint@caltech.edu

Anima Annandkumar
Computing and Mathematical Sciences
California Institute of Technology
anima@caltech.edu

William A. Goddard III
Chemistry and Chemical Engineering
California Institute of Technology
wag@caltech.edu

Abstract

Machine learning models designed for protein engineering and design typically rely on sequence-based, structure-based, or integrated representations that combine both sequence and structural information of proteins. These representations are used upon the domain knowledge: namely, that the structure (or the sequence) of a protein is partly responsible for its function. Despite their strength in capturing the evolutionary process of proteins, these representations often fall short of capturing the dynamic behavior of the protein structure. we propose a representation that incorporates knowledge of the protein's dynamic behavior obtained from molecular dynamics simulations. Our representation utilizes an unsupervised approach to observed time-series data generated from molecular dynamics simulations to encode both temporal and spatial dynamic behavior of a protein structure. Our dynamic-aware representation extracts essential relational interactions within the polymer chain revealing the interactions of sub-units of the protein which could be used to inform design strategies for protein engineering goals.

1 Introduction

In machine learning applications for protein characterization, design, and engineering, representations generally fall into one of two categories: sequence-based representations and structure-based representations. In the former, the identity of the protein is encoded using its one-dimensional representation of amino-acid or DNA codon sequence, specifying the chemical identity of the monomer units that make up the polypeptide chain, or its codon sequence, using the DNA monomer units used to encode instructions for cells to make the specific proteins. These sequence-based representations can also be designed to incorporate relationships between proteins based on evolutionary history, using methods such as multiple-sequence-alignment (MSA) or other methods of elucidating protein evolutionary lineages [1, 2]. Structure-based methods seek to represent proteins through their three-dimensional, folded structures. These structures are typically determined using experimental crystallographic or microscopic methods, such as X-ray diffraction or cryoEM. In-silico approaches to predicting protein sequence from structure are also possible [1, 3, 4], but are typically themselves a prediction based on statistical or machine-learned models.

Machine Learning for Structural Biology Workshop, NeurIPS 2024.

Most approaches that use unsupervised learning strategies on the available natural sequence data will tend to propose or perform best on sequences that stick close to those naturally occurring biological proteins, using the information explored by nature for optimizing proteins in biological contexts. However, in many protein engineering tasks, it is desirable to characterize proteins for functionality that is not present in nature. Examples include the ability to bind to novel ligands, conversion, degradation, or synthesis of novel, non-natural small molecules, or engineering properties that do not have typical selection pressure in nature, such as stability at high temperatures. It is, therefore, of interest to develop data-efficient strategies for protein-function models to aid design efforts.

Sequences as a protein representation fail to capture structural relationships between substituent parts of an individual protein. While structure-based representations account for this geometric information, they still fail to capture all the relevant physical properties and interactions that inform protein function. For example, proteins exist not as static structures, but rather as dynamic systems [5]. Static protein structures are characterizations of the minimum energy conformations of a protein chain, however, in reality, proteins exist as distributions of conformations specified by statistical physics [6]. These distributions can be encoded in dynamic equations of motion over the degrees of freedom of the protein. Computational modeling has suggested that non-trivial interactions between regions of a protein may be best captured by the effects they have on each other in their observed dynamics, particularly for long range communication pathways such as those observed in allosteric regulation in proteins [7]. These interaction pathways are not readily apparent in sequence- or structure-based representations of the protein and can only be captured by studying the system's dynamic behavior. Characterizing these dynamics in a way that reveals the relationships and directionality of interactions can lead to a more informative protein representation, enhancing efforts in protein design.

We developed an unsupervised encoder-decoder model to learn the observed dynamic structure of a protein system. The dynamics are generated from Molecular Dynamics (MD) simulations over a specified time span, resulting in a time series where each time step represents the state of the protein system. We represent the protein structure as a directed graph, with amino acids as nodes and their potential pairwise interactions as categorical edge types. We hypothesize that the topology of this graph can guide experimental design choices, enhancing protein characterization and design efforts. Once the model can encode the protein dynamic and capture the pairwise interactions between amino acids, the model enables downstream applications such as (1) detection of sites enriched in epistatic interactions relative to a parent structure [8] and (2) improved design of combinatorial libraries for directed evolution campaigns [9, 10].

2 Methods

2.1 Coarse-grained Protein Representation

To reduce the dimensionality of the observed data (and the degrees of freedom over which the learned governing equation operates), we introduce a physically motivated coarse-graining approach. The all-atom degrees of freedom are condensed into a per-residue representation, where each residue's position is defined as the center of mass of its constituent atoms, and a vector points from the alpha-carbon to the beta-carbon to indicate the side chain's orientation relative to the backbone (see Figure 1 for a visual representation). Although this coarse-grained representation may lose some atomic-level detail, it provides a consistent framework for representing different residue types, despite their varying atom counts and types. Each amino acid is represented by a vector of six real numbers: the first three are the coordinates of the center of mass, and the other three define the direction from the alpha-carbon to the beta-carbon. Thus, a protein is represented as an N by six matrix, where N is the number of residues.

2.2 Data and Simulations

The observational data that describe the dynamics of the protein are generated using MD simulations. A base protein structure of interest with a known structure from the protein data bank (PDB) is used as the starting point for all-atom MD simulations. In our experiments, the protein dynamics are simulated with a time step of two femtoseconds, under constant pressure (1 bar) and temperature (300 K), with constraints applied to hydrogen bonds. The simulations span a total of three nanoseconds, with protein structures recorded every 500 time steps. Consequently, the data are processed as time series with a time step of 1 picosecond (500×2 femtoseconds). MD simulations are conducted

using the GROMACS engine [11] and the CHARMM force field for protein systems [12, 13]. All simulations are executed on compute nodes equipped with an NVIDIA V100 (16 GB) GPU, while model training and evaluation are performed on a V100 (32 GB) GPU.

To train the model, the data are divided into time series chunks, each consisting of 50 consecutive frames (time steps). Therefore, the training data takes the form of a set of time-series samples $\mathbf{X} = \{\mathbf{x}_s\}$, where $\mathbf{x}_s = \left\{x_s^0, x_s^1, ..., x_s^T\right\}$ for time steps $t = \{0, ..., T = 50\}$. According to our proposed coarse-grained representation, the shape of x_s^t is $(N \times D)$, where N is the number of residues, and $D = 6$ is the length of the descriptor for the state of a residue at any given time t. We model each state x_s^t as a fully connected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ with $\mathcal{V}$ as residues and $\mathcal{E} = \{1\}$ for the fully connected lay out.

2.3 Model

Our model seeks to encode a time series (one dynamics) into a directed graph, which we call the interaction map $\mathcal{G}^*$. $\mathcal{G}^* = \{\mathcal{V}, \mathcal{E}^*\}$ summarizes the generative process underlying the time series where the interactions (i.e., inferred edges) are categorized into two channels $\mathcal{E}^* = \{e_1, e_2\}$.

The model consists of an encoder-decoder structure similar to that of a variational graph autoencoder [14] and follows the neural relational inference architecture as described in **(author?)** [15]. For an input system state x_j:

$$\mathbf{h}_j^1 = f_{\text{emb}}(x_j), \quad \mathbf{h}_{(i,j)}^1 = f_e^1([\mathbf{h}_i^1, \mathbf{h}_j^1]), \quad \mathbf{h}_j^2 = f_v^1\left(\sum\nolimits_{i \neq j} \mathbf{h}_{(i,j)}^1\right), \quad \mathbf{h}_{(i,j)}^2 = f_e^1([\mathbf{h}_i^2, \mathbf{h}_j^2])$$

Here, f_{emb} forms an embedding of the graph vertices $\mathcal{V}$, and f_e^1 and f_v^1 conduct two message passing rounds to produce edge embeddings $\mathbf{h}_{(i,j)}^2$ representing the interaction between node i and node j. The composite function of these layers is passed through a non-linearity in $q_\phi(\mathbf{z}_{ij} \mid \mathbf{x}) = \text{softmax}(\mathbf{h}_{(i,j)}^2)$, where ϕ represent the trainable parameters of the composed layers, and $\mathbf{z_{ij}}$ is the latent variable corresponding to edges in E^*.

The decoder uses the embedded interaction map G^* generated by the encoder to predict the next states of the system $\mathbf{x}^{t+1}$ autoregressively as $p_\theta(\mathbf{x}^{t+1} \mid \mathbf{x}^t, \mathbf{x}^{t-1}, ...\mathbf{x}^0, \mathbf{z})$. We assume the system is Markovian, so $p_\theta(\mathbf{x}^{t+1} \mid \mathbf{x}^t, \mathbf{x}^{t-1}, ...\mathbf{x}^0, \mathbf{z}) = p_\theta(\mathbf{x}^{t+1} \mid \mathbf{x}^t, \mathbf{z})$. The functional form of the decoder is as follows:

$$\tilde{\mathbf{h}}_{(i,j)}^t = \sum_k z_{ij,k} \tilde{f}_e^k([\mathbf{x}_i^t, \mathbf{x}_j^t]), \; \mu_j^{t+1} = \mathbf{x}_j^t + \tilde{f}_v\left(\sum\nolimits_{i \neq j} \tilde{\mathbf{h}}_{(i,j)}^t\right), \; p(\mathbf{x}_j^{t+1} \mid \mathbf{x}^t, \mathbf{z}) = \mathcal{N}\left(\mu_j^{t+1}, \sigma^2 \mathbf{I}\right)$$

Note that there are two channels for the edges as specified in the indexing of $z_{ij,1}$ and $z_{ij,2}$. To avoid degenerate decoders that could be encouraged by a single step reconstruction term in the ELBO loss function, we compute the loss for reconstructing multiple time steps into the future as well as implementing separate MLPs for each edge embedding channel [15].

We categorize each potential interaction between coarse-grained beads into two channels. The decoder skips the first learned edge, while both edges are processed through a Gumbel-Softmax function [16]. This allows the fraction of weight assigned to the first, skipped edge to be interpreted as the model downplaying the importance of the interaction represented by that edge [17]. In our experiments, we focus on analyzing the weight of the second edge, which reflects the importance of the directed interaction between two coarse-grained beads within our observed dynamic system.

3 Experiments

3.1 Setup

Calmodulin is a calcium ion-sensing protein that is highly conserved and present in eukaryotic cells [18]. Its structure was determined and refined to 1.7 Å resolution [19, 20]. The protein structure consists of two globular regions connected by a helix structure. Four distinct Ca^{2+} sites exist in the globular regions with differential affinity to calcium ion binding. Experimental studies have shown that a coupling between the sites must exist, including a conformational change that occurs upon binding for some of the sites which allows the protein to enter an active form responsible for binding a variety of other proteins [21, 22]. For our experiment, we generate dynamics data as described in section 2.2 for the apo-calmodulin system and train our model to uncover the latent interaction graph.

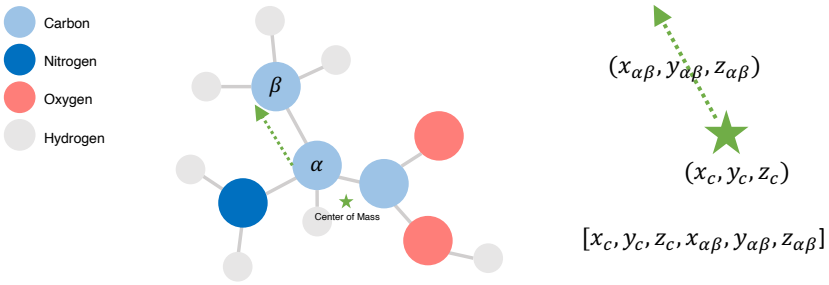

Figure 1: All atom schematic depiction of an amino acid and its coarse-grained representation.

3.2 Results

We generated three nanoseconds of unbiased molecular dynamics data for the calmodulin system as described in section 2.2. The resulting trajectories of the protein atoms were then coarse-grained as described in section 2.1 and used to train the model architecture.

After training, the encoder is used to generate the interaction maps across time series to summarize the observed trajectories. The average of the interaction maps is computed across the entire three nanoseconds span and is shown in Figure 2. Using the interaction map, we draw several qualitative inferences about various sub-domains on the protein structure. For example, there exists an asymmetric interaction between a domain on one end of the protein (green) and a separate domain on the other end (purple). After training, the model has learned that, in order to predict the dynamics of the latter domain (purple), the positions of the former domain (green) are informative, indicated by the large weight placed on the connected edge. In contrast, the reverse does not follow; the model has learned that, in order to predict the motion of the first domain (green), information from the second (purple) domain is not informative. From this, we can interpret the model as suggesting that the dynamics of the purple domain are more conditionally dependent on those of the green domain relative to the vice-versa.

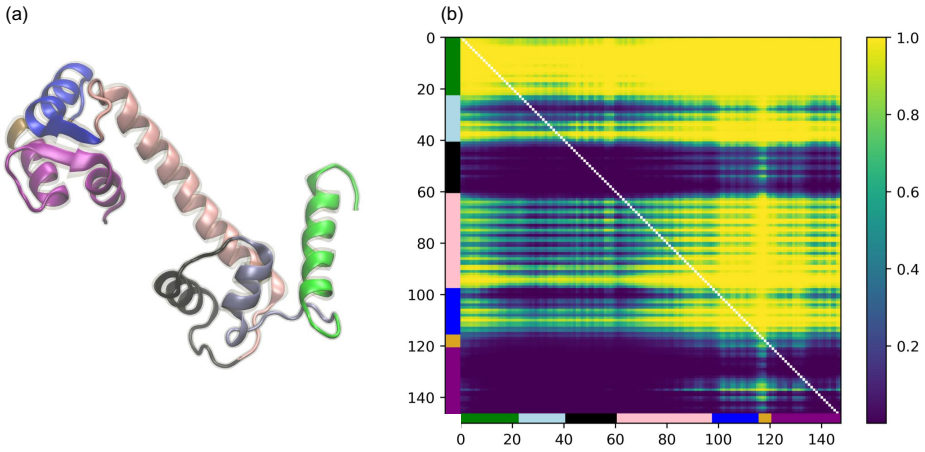

Figure 2: (a): Calmodulin backbone and sub-domains in different colors. (b) The interaction map across three nanoseconds observed the dynamic of the structures. The partitions are colored to represent the sub-domains. The edge value ranges from 0 to 1 and corresponds to the fraction of weight assigned to the non-zero interaction in the 2-edge trained model.

4 Conclusion and Future Work

We proposed a method to capture the dynamic behavior of protein structures and generate more informative representations. While we employed a Graph Neural Network (GNN) architecture to model neural relational inference, graph transformer models [23] could be used to leverage global attention across the entire protein structure. To handle the time series nature of the data, such models

would need to be integrated with temporal encoders, such as Temporal Fusion Transformers (TFTs) [24], enabling more effective processing of dynamic information.

Furthermore, one can explore more effective ways to determine the directed causal relationships between interacting variables in our observed systems by employing interventional strategies. This includes introducing point mutations to parent protein sequences and modifying force-field parameters to isolate specific interactions at the all-atom resolution, allowing us to observe the resulting changes in the model's learned dynamic encodings.

On the application side, the inferred interaction maps can inform protein engineering and design strategies. One potential application is providing actionable insights for site selection in targeted libraries for directed evolution (DE). For instance, one can explore whether the pairwise interaction mappings generated by our trained models can infer pairwise epistasis between residues in a protein structure, which could be leveraged to design more effective combinatorial libraries for site-saturation mutagenesis in DE campaigns. This will be examined by comparing the dynamic representations from our models with experimental data from deep mutational scanning.

References

[1] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

[2] Ali Madani, Ben Krause, Eric R Greene, Subu Subramanian, Benjamin P Mohr, James M Holton, Jose Luis Olmos, Caiming Xiong, Zachary Z Sun, Richard Socher, et al. Large language models generate functional protein sequences across diverse families. *Nature Biotechnology*, 41(8):1099–1106, 2023.

[3] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

[4] Carol A Rohl, Charlie EM Strauss, Kira MS Misura, and David Baker. Protein structure prediction using rosetta. In *Methods in enzymology*, volume 383, pages 66–93. Elsevier, 2004.

[5] Katherine Henzler-Wildman and Dorothee Kern. Dynamic personalities of proteins. *Nature*, 450(7172):964–972, 2007.

[6] Hans Frauenfelder, Stephen G Sligar, and Peter G Wolynes. The energy landscapes and motions of proteins. *Science*, 254(5038):1598–1603, 1991.

[7] Kateri H DuBay, Gregory R Bowman, and Phillip L Geissler. Fluctuations within folded proteins: implications for thermodynamic and allosteric regulation. *Accounts of chemical research*, 48(4):1098–1105, 2015.

[8] Tyler N Starr and Joseph W Thornton. Epistasis in protein evolution. *Protein science*, 25(7):1204–1218, 2016.

[9] Victor Sayous, Paul Lubrano, Yanyan Li, and Carlos G Acevedo-Rocha. Unbiased libraries in protein directed evolution. *Biochimica et Biophysica Acta (BBA)-Proteins and Proteomics*, 1868(2):140321, 2020.

[10] Feng Cheng, Leilei Zhu, and Ulrich Schwaneberg. Directed evolution 2.0: improving and deciphering enzyme properties. *Chemical Communications*, 51(48):9760–9772, 2015.

[11] Mark James Abraham, Teemu Murtola, Roland Schulz, Szilárd Páll, Jeremy C Smith, Berk Hess, and Erik Lindahl. Gromacs: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, 1:19–25, 2015.

[12] Bernard R Brooks, Charles L Brooks III, Alexander D Mackerell Jr, Lennart Nilsson, Robert J Petrella, Benoît Roux, Youngdo Won, Georgios Archontis, Christian Bartels, Stefan Boresch, et al. Charmm: the biomolecular simulation program. *Journal of computational chemistry*, 30(10):1545–1614, 2009.

[13] Alexander D Mackerell Jr, Michael Feig, and Charles L Brooks III. Extending the treatment of backbone energetics in protein force fields: Limitations of gas-phase quantum mechanics in reproducing protein conformational distributions in molecular dynamics simulations. *Journal of computational chemistry*, 25(11):1400–1415, 2004.

[14] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[15] Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *International conference on machine learning*, pages 2688–2697. PMLR, 2018.

[16] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[17] CWJ Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, page 37, 1969.

[18] D Brent Halling, Benjamin J Liebeskind, Amelia W Hall, and Richard W Aldrich. Conserved properties of individual ca2+-binding sites in calmodulin. *Proceedings of the National Academy of Sciences*, 113(9):E1216–E1225, 2016.

[19] Y Sudhakar Babu, Charles E Bugg, and William J Cook. Structure of calmodulin refined at 2.2 å resolution. *Journal of molecular biology*, 204(1):191–204, 1988.

[20] Rajagopal Chattopadhyaya, William E Meador, Anthony R Means, and Florante A Quiocho. Calmodulin structure refined at 1.7 å resolution. *Journal of molecular biology*, 228(4):1177–1192, 1992.

[21] Mladen Milos, Jean-Jacques Schaer, Michelle Comte, and Joseph A Cox. Evidence for four capital and six auxiliary cation-binding sites on calmodulin: divalent cation interactions monitored by direct binding and microcalorimetry. *Journal of inorganic biochemistry*, 36(1):11–25, 1989.

[22] Mitsuhiko Ikura, G Marius Clore, Angela M Gronenborn, Guang Zhu, Claude B Klee, and Ad Bax. Solution structure of a calmodulin-target peptide complex by multidimensional nmr. *Science*, 256(5057):632–638, 1992.

[23] Seongjun Yun, Minbyul Jeong, Raehyun Kim, Jaewoo Kang, and Hyunwoo J Kim. Graph transformer networks. *Advances in neural information processing systems*, 32, 2019.

[24] Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4):1748–1764, 2021.