Predicting directional flexibility in proteins

Vsevolod Viliuga [*1,2] **Leif Seute** [2,3,4] **Matteo Tadiello** [1] **Nicolas Wolf** [2,3,4]
Frauke Gräter [2,4] **Arne Elofsson** [1]

[1] DBB and SciLifeLab at Stockholm University, Stockholm, Sweden
[2]Max Planck Institute for Polymer Research, Mainz, Germany
[3]Heidelberg Institute for Theoretical Studies, Heidelberg, Germany
[4]IWR, Heidelberg University, Heidelberg, Germany

Abstract

Predicting protein dynamics is a long-standing problem in computational structural biology. Often, protein function critically depends on local directed motions, such as hinge movements, catalytic loop rearrangements and domain reorientations, which can be characterized by directional flexibility and correlated structural motions of the protein backbone. While Molecular Dynamics (MD) simulations provide an established but often prohibitively expensive approach, recent deep generative models aim to reduce this cost by directly predicting conformational ensembles, emulating MD. However, due to their large size and the need to generate several states until the derived dynamical properties converge, these models remain expensive. In this work, we propose BackFlip-2: a fast SE(3)-equivariant graph neural network trained to directly predict dynamical descriptors, such as directional backbone flexibility and pairwise dynamic correlations, from an equilibrium structure. In a series of experiments, we show that our model matches the accuracy of substantially larger ensemble generation models while being orders of magnitude faster, and demonstrate that the proposed equivariant architecture is especially well-suited for capturing anisotropic motions in proteins. BackFlip-2 model weights, training and inference code are available at `https://github.com/graeter-group/backflip`.

1 Introduction

Proteins are essential biomolecules that carry out a wide range of functions, ranging from DNA replication and energy metabolism to sophisticated coordinated processes such as chromosome segregation and cell division [1]. Since the three-dimensional structure of a protein determines much of its function, accurately describing structural features is central to understanding its biological role. Importantly, proteins are not static objects: their function can depend on both large conformational changes occurring on timescales beyond milliseconds, but also on local dynamical degrees of freedom explored within hundreds of nanoseconds. Such local, fast motions can coordinate protein assembly and oligomerization [2], shape active site geometry [3–5] and determine substrate specificity [6]. In turn, larger and typically slower loop, hinge, and domain motions drive catalysis [7, 8, 3], molecular recognition [9, 10], and allosteric regulation [11]. Importantly, many of these functional motions are anisotropic [12, 13], i.e. the protein structural elements involved in the function move along a preferred direction. For example, in ionotropic glutamate receptors, ligand-binding domain closure is coupled to channel gating [14], while coordinated directional LID and NMPbind domain movements in adenylate kinase guide catalysis [15, 16]. Thus, understanding protein function requires

*Corresponding author. Contact: vsevolod.viliuga@scilifelab.se

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

a description of protein dynamics across different timescales, capturing directionality and collective motions.

An established way to characterize protein dynamics is to run Molecular Dynamics (MD) simulations. In MD, a molecular system is evolved in time by numerically integrating Newton's equations of motion. However, numerical stability requires femtosecond-scale integration time steps, while biologically relevant processes often emerge only over much longer timescales, rendering MD computationally expensive and hence limiting its scalability to large sets of proteins. A faster physics-based alternative is Normal Mode Analysis (NMA), which describes protein motions around an equilibrium structure assuming an effective harmonic potential [17]. However, this approximation is restricted to strictly local fluctuations [18, 19] and often suffers from limited accuracy [20].

Ensemble generation models Recently, several deep generative models based on diffusion [21] and flow matching [22, 23] have been proposed to obtain protein conformational ensembles at reduced cost by emulating MD simulation. Reflecting the importance of dynamics across timescales, different models have targeted different time horizons. For example, BioEmu [24] focuses on long, millisecond-scale conformational ensembles, while AlphaFlow [25], BBFlow [26], and related methods [27, 28] model MD ensembles on shorter, sub-microsecond timescales by training on the ATLAS dataset of 300 ns MD trajectories [29]. These approaches accurately emulate MD-like conformational ensembles, but this comes at a cost: the ensemble generation models are typically large and require the sampling of multiple conformations to obtain converged dynamical observables, with inference times ranging from several minutes to hours for a single protein.

Predicting dynamical descriptors directly In a complementary line of work, structural flexibility is predicted directly, avoiding ensemble generation altogether. Models such as Flexpert [30], Pegasus [31], and BackFlip [32] predict MD-derived per-residue flexibility from sequence or structure. However, these models predict scalar isotropic flexibility (e.g. RMSF). Yet, most of the functional motions are characterized not only by the extent of residue movements, but also by their direction and by residue-residue correlations [12]. The recently proposed DynaProt [33] attempts to address this limitation by predicting per-residue covariance matrices describing anisotropic flexibility, together with pairwise residue couplings. However, the covariance matrices predicted by DynaProt are SE(3)-invariant, which is an important drawback: covariance matrices derived from MD trajectories are not invariant but transform according to the matrix representation of $R \in \mathrm{SO}(3)$ under global rotation.

Main contributions In this work, we propose BackFlip-2: an SE(3)-equivariant neural network for predicting per-residue directional flexibility on the 300-500 ns timescale regime from the protein backbone of an equilibrium structure. BackFlip-2 predicts anisotropic covariance matrices and pairwise residue correlations, generalizing our previous direction agnostic model BackFlip [32]. The proposed model achieves state-of-the-art performance in both accuracy and runtime. It outperforms directly comparable flexibility prediction baselines and is even on par with substantially larger ensemble generative models, such as AlphaFlow [25], while being over four orders of magnitude faster. We achieve this by constructing a lightweight model based on the IPA algorithm from AlphaFold2 [34] for predicting covariances that transform under the matrix representation of $\mathrm{SE}(3)$, and scalar pairwise residue couplings. We train and evaluate our model on the well-established ATLAS dataset [29] comprising MD simulations for 1390 proteins each covering in total 300 ns, and further evaluate it on mdCATH [35] and *de novo* proteins. Our ablations demonstrate that due to the proposed equivariance, the model performs especially well at capturing highly anisotropic residue motions with directional specificity, which are of particular interest in practice.

1.1 Related Work

MD simulation datasets The ATLAS dataset [29] provides standardized MD simulations for 1390 structurally non-redundant proteins, each with three independent 100 ns replicas obtained with the CHARMM36m force-field at 300 K and a 2 fs integration time step [29]. The mdCATH dataset [35] provides MD simulations for 5398 protein domains taken from the CATH database [36] with five trajectories per domain simulated at different temperatures (320, 348, 379, 413, and 450 K) covering 500 ns in total, using the CHARMM22 force field and a 4 fs integration time step.

Predicting protein dynamics An early approach to predict protein dynamics faster than MD is Normal Mode Analysis (NMA), which approximates local perturbations of the protein by an effective harmonic potential, limiting it to small-amplitude motions [18, 19]. Flow- and diffusion-based generative models instead learn conformational ensembles from MD data: In AlphaFlow [25], the protein structure prediction model AlphaFold2 [34] is fine-tuned as a flow matching model on MD trajectories from ATLAS, significantly outperforming NMA and also the training-free MSA subsampling approach proposed by Wayment-Steele et al. [37]. BBFlow [26] formulates ensemble generation as a structure-to-structure flow matching task and is substantially faster than AlphaFlow.

To directly predict RMSF, several model have been proposed previously (see also App. A.4): FlexPert [30] and Pegasus [31] predict per-residue RMSF from the sequence using pretrained protein language models, while the faster BackFlip [32] predicts isotropic RMSF from the protein structure using a graph neural network. The recent model DynaProt [33] generalizes the idea from BackFlip to predict flexibility based on the protein backbone structure beyond RMSF and proposes to predict per-residue 3×3 covariance matrices and pairwise residue couplings with an invariant graph neural network.

2 Per-residue covariances and pairwise couplings as directional flexibility

We aim to learn two observables from MD simulations characterizing residue-wise dynamical degrees of freedom and residue correlations: *per-residue covariances* $\Sigma^{(ii)} \in \mathbb{R}^{3\times3}$, which describe anisotropic fluctuations of residue i, and *pairwise scalar couplings* $C \in \mathbb{R}^{N\times N}$, which capture correlated motions between residue pairs. We will define these quantities below.

Full covariance matrix To compute $\Sigma^{(ii)}$ and C from MD trajectories, we first superpose all trajectory frames onto an *equilibrium reference conformation* to remove global translation and rotation. The full positional covariance Σ_{full} is then defined as covariance of the stacked positions $x \in \mathbb{R}^{3N}$,

$$\Sigma_{\text{full}} \equiv \big\langle\, (x - \langle x\rangle) \otimes (x - \langle x\rangle) \,\big\rangle \in \mathbb{R}^{3N\times3N}, \tag{1}$$

where $\langle\rangle$ denotes the ensemble average and $\otimes$ is the tensor product.

The full covariance matrix can be decomposed into N^2 individual 3×3 matrices describing the correlation between residues i and j,

$$\Sigma^{(ij)} = \big\langle\, (x_i - \langle x_i\rangle) \otimes (x_j - \langle x_j\rangle) \,\big\rangle \in \mathbb{R}^{3\times3}. \tag{2}$$

Crucially, under global rotations $R \in \mathrm{SO}(3)$ of the reference structure, these matrices transform under the matrix representation of $\mathrm{SO}(3)$:

$$\Sigma^{(ij)} \mapsto \big\langle\, (R\,x_i - \langle R\,x_i\rangle) \otimes (R\,x_j - \langle R\,x_j\rangle) \,\big\rangle = R\,\Sigma^{(ij)} R^T, \tag{3}$$

Per-residue covariance We then define the *per-residue covariance* Σ as subset of the covariance matrices in Eq. 2, $\Sigma \equiv \{\Sigma^{(ii)}\}_{i=1}^{N}$, which can be interpreted as measure for directional flexibility (anisotropy) of a residue (visualization in Fig. 1). From these per-residue covariances, one can obtain the scalar *root-mean-square fluctuations* (RMSF) by taking the trace,

$$\mathrm{RMSF}_i = \sqrt{\left\langle \sum_{d=1}^{3} \left(x_{i,d} - \langle x_{i,d}\rangle\right)^2 \right\rangle} = \sqrt{\sum_{d=1}^{3} \left\langle \left(x_{i,d} - \langle x_{i,d}\rangle\right)^2 \right\rangle} = \sqrt{\sum_{d=1}^{3} \Sigma^{(ii)}_{dd}} = \sqrt{\mathrm{tr}\big(\Sigma^{(ii)}\big)}, \tag{4}$$

where we used the linearity of the ensemble average operator $\langle\,\rangle$ in the second equality. In this sense one can think of the per-residue covariance as a generalization of the established scalar RMSF to quantify the anisotropy of fluctuations. Note that the RMSF does not change under global rotations since the trace is $\mathrm{O}(n)$ invariant,

$$\forall R \in \mathrm{O}(n),\ \forall A \in \mathbb{R}^{n\times n}: \quad \mathrm{tr}\big(RAR^T\big) = \mathrm{tr}(A). \tag{5}$$

Pairwise residue couplings We define the *pairwise residue couplings* between residues i and j as the trace of the off-diagonal block matrices $\Sigma^{(ij)}$,

$$C_{ij} \equiv \mathrm{tr}\left(\Sigma^{(ij)}\right) = \big\langle\, (x_i - \langle x_i\rangle) \cdot (x_j - \langle x_j\rangle) \,\big\rangle, \tag{6}$$

that is as the correlation of the relative displacements of both residues across the ensemble. These couplings hence indicate how likely it is that two residues move in the same direction (positive values of C_{ij}) or antiparallel (negative values of C_{ij}). Importantly, the trace is rotation invariant (Eq. 5), which makes the couplings independent of global rotations of the reference structure. This is in contrast to the definition in [33], where the mean is used instead of the trace, making the couplings dependent on the absolute orientation of the reference structure. From C, one can compute the established *dynamic cross-correlation matrix* (DCCM) $\tilde{\mathrm{C}}$ by normalizing each entry with the corresponding per-residue fluctuation magnitudes:

$$\tilde{\mathrm{C}} \in \mathbb{R}^{N\times N}, \qquad \tilde{\mathrm{C}}_{ij} = \frac{\mathrm{C}_{ij}}{\sqrt{\mathrm{C}_{ii}\,\mathrm{C}_{jj}}} = \frac{\mathrm{C}_{ij}}{\mathrm{RMSF}_i\,\mathrm{RMSF}_j}. \tag{7}$$

The resulting entries satisfy $\tilde{\mathrm{C}}_{ij} \in [-1, 1]$ and quantify the correlation of displacement directions.

3 Method

We propose an $\mathrm{SE}(3)$-equivariant network that predicts (i) equivariant per-residue covariances and (ii) scalar pairwise residue couplings, as derived from MD simulations. The model relies on the protein backbone geometry and does not require evolutionary sequence information or pre-trained protein language model embeddings, enabling orders of magnitude faster inference at improved accuracy. It consists of two components: a backbone encoder and a dynamics prediction module that receives learned backbone representations. For a structure with N residues, the model jointly predicts

$$(\hat{\Sigma}, \hat{C}) \in \mathbb{R}^{N\times 3\times 3} \times \mathbb{R}^{N\times N}, \tag{8}$$

with the per-residue covariance Σ and pairwise coupling C as defined above. For training, we define a composite objective $\mathcal{L} = \mathcal{L}_\Sigma + \mathcal{L}_C$ (see Eq. 13 and 15). In the following, we describe the model architecture in detail.

3.1 Backbone encoder

We follow the well-established representation of a protein backbone as a set of rigid body poses $T \in \mathrm{SE}(3)^N$, where each residue frame is given by $T_i = (x_i, R_i)$ with translational component $x_i \in \mathbb{R}^3$ and rotational component $R_i \in \mathrm{SO}(3)$. To learn an expressive representation of protein backbone geometry, we make use of the Invariant Point Attention (IPA) introduced in AlphaFold2 [34]. The intermediate residue features $h \in \mathbb{R}^{N\times d}$ express each node embedding h_i in the local coordinate system defined by the residue frame T_i. Upon message passing, the node features are transformed into the coordinate frame of the target residue, as in AlphaFold2. This ensures $\mathrm{SE}(3)$ equivariance under arbitrary node-wise learnable layers, and also allows the model to capture geometric information about relative displacements and orientations between residues. We initialize h_i by combining an amino acid type one-hot-encoding with a positional encoding of the index in the protein chain, followed by a projection to the latent dimension d. Initial edge features $e \in \mathbb{R}^{N\times N\times c}$ are defined for all residue pairs by combining a relative positional index encoding with the pairwise distance between the corresponding residues. Pairwise C_α distances are binned uniformly between 0 and 20 Å. Node and edge embeddings are refined by a stack of l modified IPA blocks. Each block constructs $\mathrm{SE}(3)$-invariant attention scores from point-valued query and key features, followed by a transformer acting on the scalar node features [34]. Unlike structure prediction models, the backbone frames are held fixed, i.e. the backbone update block is omitted, only node and edge embeddings are updated in each layer. After l blocks, both the node embeddings $h^{(l)}$ and edge embeddings $e^{(l)}$ are passed to the prediction heads described below. We set $d = 96$, $c = 64$, and $l = 4$.

3.2 Dynamics prediction module

We propose a dynamics prediction module that maps the learned backbone node and edge embeddings to per-residue covariances and pairwise residue couplings, respectively. A key architectural difference to the recently proposed DynaProt [33] is that we predict $\mathrm{SE}(3)$-equivariant per-residue covariance matrices according to the transformation behavior derived in Eq. 3, whereas DynaProt proposes an $\mathrm{SE}(3)$-invariant covariance head. In our experiments we demonstrate that the $\mathrm{SE}(3)$ equivariance is crucial for capturing anisotropies in the covariances (Fig. 1 and Sec. 4.2). Beyond this, our dynamics

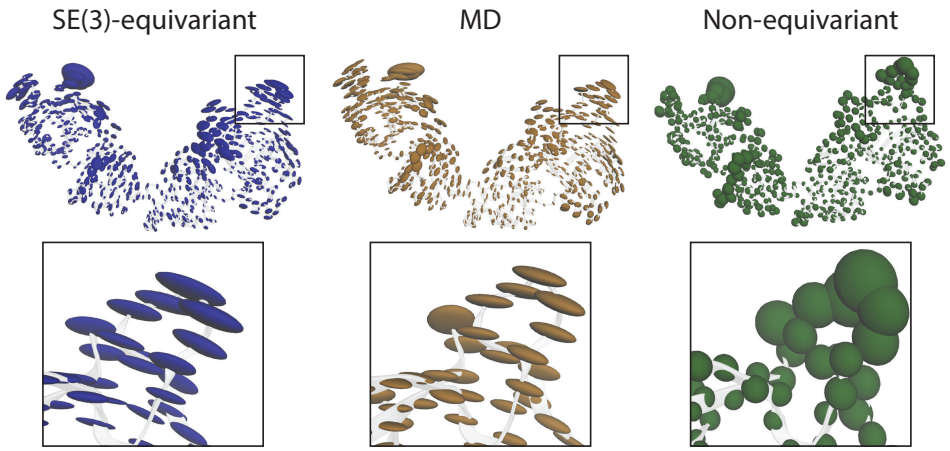


Figure 1: Equivariance is crucial to capture anisotropic covariances. We visualize ellipsoids representing per-residue covariance matrices for the ATLAS test set protein 7asgA predicted with our proposed $\mathrm{SE}(3)$-equivariant model (left), the MD-derived ground truth (middle) and the prediction of the non-equivariant model variant from Sec. 4.2 (right).

prediction module leverages the edge embeddings $e^{(l)}$ returned by the backbone encoder for the prediction of pairwise residue couplings, which encode pairwise geometric information and interaction strengths between residues directly. DynaProt instead predicts both quantities from node embeddings alone, and introduces a separate pair-attention module to capture inter-residue dependencies. The use of edge encodings eliminates the need for a heavy pair-attention module, substantially reducing parameter count while also improving predictive performance, as we demonstrate in our experiments. We describe both heads in detail below.

3.2.1 Equivariant per-residue covariance head

To predict the per-residue covariance matrices $\Sigma^{(ii)}$, we construct an output head that satisfies two constraints by design: symmetric positive semi-definiteness, and $\mathrm{SE}(3)$ equivariance. We use this as inductive bias since the ground truth covariances satisfy both of these constraints (Eq. 2 and Eq. 3).

Symmetric positive semi-definiteness We employ an Multi-Layer Perceptron (MLP) that maps the node embedding $h_i^{(l)}$ to a matrix $A^{(i)} \in \mathbb{R}^{3\times 3}$, from which we construct the predicted local-frame covariance matrix as

$$\hat{\Sigma}^{(ii)}_{\mathrm{loc}} = A^{(i)} {A^{(i)}}^{\top}, \tag{9}$$

which by construction is symmetric positive semi-definite (SPSD) for any matrix $A^{(i)}$. We compare our approach to the Cholesky-factorization method from DynaProt in App. A.1.

SE(3) equivariance Since the target blocks $\Sigma^{(ii)}$ computed from MD trajectories transform equivariantly under global rotations (Eq. 3), the predicted covariance should also satisfy this behavior. In order to obtain an equivariant predicted covariance matrix, we rotate the local covariance into the global frame using the residue rotation $R_i \in \mathrm{SO}(3)$,

$$\hat{\Sigma}^{(ii)} = R_i \, \hat{\Sigma}^{(ii)}_{\mathrm{loc}} \, R_i^{\top}. \tag{10}$$

Under a global rotation $R \in \mathrm{SO}(3)$, the residue frames transform as $R_i \mapsto R R_i$, while $\hat{\Sigma}^{(ii)}_{\mathrm{loc}}$ is unchanged since it is defined locally. The predicted global-frame covariance therefore transforms as

$$\hat{\Sigma}^{(ii)} \;\mapsto\; (RR_i)\, \hat{\Sigma}^{(ii)}_{\mathrm{loc}} \, (RR_i)^{\top} = R \, \hat{\Sigma}^{(ii)} \, R^{\top}, \tag{11}$$

matching the required equivariance from Eq. 3. This rotated matrix is also SPSD by definition since

$$\forall v \in \mathbb{R}^3 : \quad v^{\top} \hat{\Sigma}^{(ii)} v = v^{\top} \left(R_i \, \hat{\Sigma}^{(ii)}_{\mathrm{loc}} \, R_i^{\top} \right) v = (R_i^{\top} v)^{\top} \hat{\Sigma}^{(ii)}_{\mathrm{loc}} \, (R_i^{\top} v) \geq 0, \tag{12}$$

where we used SPSD of $\hat{\Sigma}^{(ii)}_{\mathrm{loc}}$ in the last step. Hence the constructed learnable covariance head $\hat{\Sigma}^{(ii)}$ satisfies both of the constraints proposed above. We show in our ablation study in Sec. 4.5 that both of these constraints are indeed favorable in practice, especially for anisotropic covariances (Fig. 1).

Loss function As in DynaProt [33], we train the covariance head with a log-Frobenius loss between the predicted and ground-truth covariance matrices, given by

$$\mathcal{L}_{\Sigma} = \left\| \log\left(\hat{\Sigma}^{(ii)}\right) - \log\left(\Sigma^{(ii)}\right) \right\|_F^2, \tag{13}$$

where $\log(\cdot)$ denotes the matrix logarithm and $\|\cdot\|_F$ the Frobenius norm.

3.2.2 Pairwise residue couplings head

The head for predicting pairwise residue couplings takes as input the backbone edge representation $e^{(l)} \in \mathbb{R}^{N\times N\times c}$ and predicts a pairwise scalar coupling matrix $\hat{C} \in \mathbb{R}^{N\times N}$. The target couplings computed from MD trajectories are invariant under global rotations (Eq. 6), hence we construct the predicted couplings from the edge embeddings $e_{ij}^{(l)}$, which are $\mathrm{SE}(3)$-invariant scalar features. For each residue pair (i, j), we directly pass the edge embedding $e_{ij}^{(l)}$ through an MLP to predict $\hat{C}_{ij}$. We symmetrize the predicted coupling matrix as $\hat{C} = (\hat{C} + \hat{C}^\top)/2$ to ensure that the predicted coupling matrix is symmetric.

Loss function We train the coupling head with a composite loss, with the primary objective being the correlation matrix distance (CMD) loss [38]:

$$d_{\mathrm{corr}}(\hat{C}, C) = 1 - \frac{\mathrm{tr}(\hat{C}\, C)}{\|\hat{C}\|_F \, \|C\|_F}. \tag{14}$$

CMD measures the similarity between two coupling matrices through their normalized Frobenius inner product. However, because it is scale-invariant, we complement it with an MAE term on the predicted coupling values to explicitly enforce the correct magnitude. In addition, we analytically compute the normalized DCCM as in Eq. 7 induced by the predicted pairwise scalar coupling matrix and apply an auxiliary MSE loss on it. The composite loss function is then given as

$$\mathcal{L}_C = d_{\mathrm{corr}}(\hat{C}, C) + \lambda_{\mathrm{MAE}}\, \mathrm{MAE}(\hat{C}, C) + \lambda_{\mathrm{DCCM}}\, \mathrm{MSE}\left(\hat{\tilde{C}}, \tilde{C}\right), \tag{15}$$

where $\hat{C}$ denotes the predicted and C the ground-truth coupling matrix, $\hat{\tilde{C}}$ and $\tilde{C}$ the corresponding normalized DCCMs, and $\lambda_{\mathrm{MAE}}, \lambda_{\mathrm{DCCM}}$ are hyperparameters denoting loss weights.

4 Experiments

Training We train BackFlip-2 on the ATLAS dataset using both the AlphaFlow [25] and FlexPert [30] data splits and compare against baselines on their respective split for fairness. We train the model and all ablation variants *jointly* on both per-residue covariance and pairwise coupling tasks for 200 epochs, requiring approximately 4 GPU hours on a single NVIDIA A100 GPU, and select the checkpoints based on the lowest validation loss. We additionally construct and train a model on a combined ATLAS/mdCATH dataset to assess the effect of joint training on MD data obtained under different simulation settings, following the same procedure as for ATLAS (details in App. A.3). All datasets along with the training, validation and test splits, and model weights are available at `https://github.com/graeter-group/backflip`.

Metrics We evaluate how well the model predicts three MD-derived dynamical descriptors: (i) per-residue covariances (Eq. 2), (ii) RMSF derived from the predicted covariances via Eq. 4, and (iii) pairwise residue couplings (Eq. 6). For per-residue covariance, we assess accuracy of the predictions by reporting the variance contribution of the root mean Wasserstein distance (RMWD) between predicted and MD covariance matrices, together with the variance contribution to the symmetric KL divergence, both proposed in AlphaFlow [25]. To explicitly evaluate the directionality of the predicted covariances, we construct ellipsoids from the predicted and MD per-residue covariances and report their volume overlap using IoU and Dice metrics, as described in App. A.2. For RMSF, we report the per-target Pearson correlation coefficient r and mean absolute error (MAE) to the ground truth. For evaluating the pairwise residue couplings, we reconstruct the induced dynamic cross-correlation matrices (DCCM, Eq. 7) as in [26] and report Pearson correlation and MAE over all residue pairs. All ground-truth quantities are computed from three stacked MD trajectories.

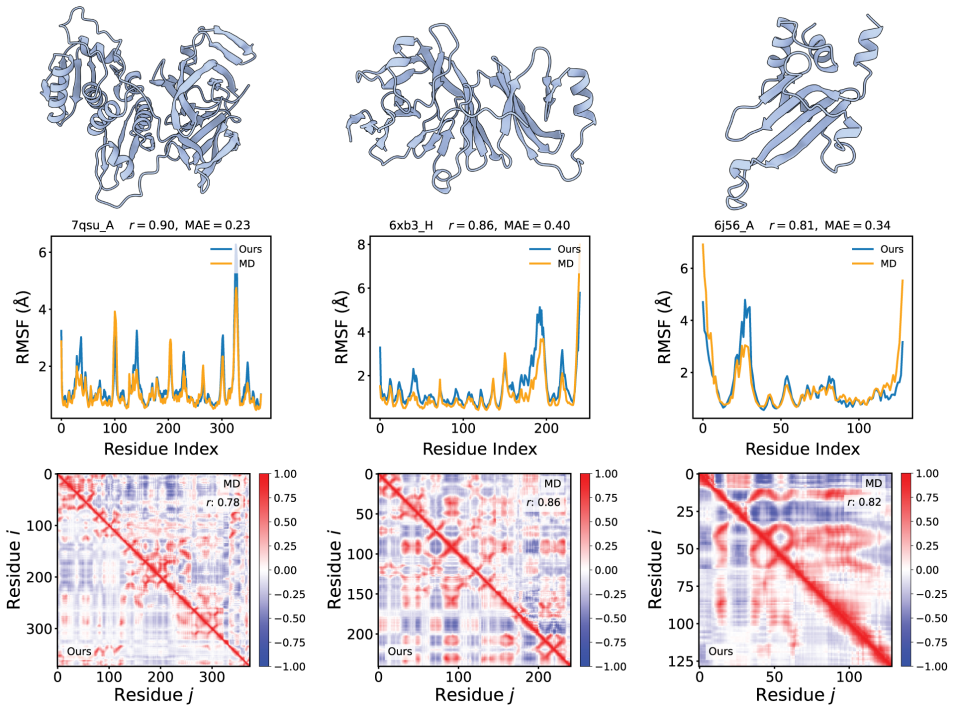


Figure 2: Predictions of per-residue RMSF and pairwise residue couplings (DCCM) made by BackFlip-2 for three proteins from the ATLAS test set compared to MD. For the DCCM matrices, lower triangle of the matrix is the prediction, and upper is MD.

Baselines For the RMSF prediction task, we compare BackFlip-2 to FlexPert, Pegasus, its predecessor BackFlip, DynaProt, AlphaFlow-MD-Templates (AFMD-T), BBFlow, and BioEmu (Sec. 1.1). FlexPert, Pegasus and BackFlip are trained to predict RMSF directly whereas our model and DynaProt predict per-residue covariance, from which we compute RMSF analytically via Eq. 4. AFMD-T, BBFlow, and BioEmu generate conformational ensembles, from which we compute RMSF. For per-residue covariance and residue couplings prediction, we compare our model to DynaProt, AFMD-T, BBFlow, and BioEmu. We note that BioEmu was not trained on the standardized ATLAS dataset, but generates ensembles corresponding to MD-timescales way beyond the 300 ns timescale of ATLAS (Sec. 1.1), explaining its poor performance if evaluated on ATLAS. We include it for completeness. For DynaProt, neither code or model weights are published at the time of submission, hence we can only compare the values reported in the paper.

4.1 ATLAS benchmark performance

Predicting RMSF The proposed model outperforms RMSF prediction models FlexPert, Pegasus and the predecessor BackFlip on RMSF prediction task. On the held-out ATLAS test set, it achieves higher per-target correlation and lower MAE with respect to MD than all baselines (Tab. 1). Compared to FlexPert and Pegasus, BackFlip-2 is also at least an order of magnitude faster and uses substantially fewer trainable parameters. BackFlip is similarly efficient, but remains less accurate. In addition to the ATLAS test set, we evaluate RMSF prediction on 100 *de novo* proteins generated with RFdiffusion [39] and FrameFlow [40] and simulated under the same settings as ATLAS (made available in [26]). Importantly, our proposed model remains robust for *de novo* proteins: it also achieves state-of-the-art performance on this dataset. The performance of FlexPert and Pegasus, which rely on protein Language Model embeddings, however, deteriorates substantially. We visualize RMSF predictions for three randomly selected ATLAS test proteins in Fig. 2.

Predicting per-residue covariances and pairwise couplings We next compare BackFlip-2 to DynaProt, which is trained to predict the same observables, and find that our model performs substantially better across all evaluated metrics. In particular, our model achieves higher accuracy for

Table 1: BackFlip-2 achieves state-of-the-art performance among RMSF prediction models. We evaluate all models on the ATLAS test set (choosing the FlexPert split [30] for a fair comparison) and on 100 *de novo* proteins. We report the per-target Pearson correlation r and mean absolute error (MAE) between predicted and ground-truth RMSF. Metrics are reported as *mean* over proteins in each test set with deviation computed by bootstrapping samples 1000 times. For MD, we compare individual trajectories. Inference time is reported for an example protein of length 300. Best values are in bold, second-best underlined.

	ATLAS test set		De novo proteins		# Params	Time [s]
	Per-target r (↑)	MAE [Å] (↓)	Per-target r (↑)	MAE [Å] (↓)		
MD (Ground Truth)	0.88 (0.01)	0.47 (0.00)	0.84 (0.02)	0.19 (0.01)	–	$\mathcal{O}(10,000)$
FlexPert†	0.83 (0.01)	0.80 (0.02)	0.57 (0.04)	0.67 (0.02)	1.2B	0.4
Pegasus†	0.75 (0.01)	0.82 (0.02)	0.65 (0.03)	0.74 (0.01)	11M	2.5
BackFlip	<u>0.84</u> (0.01)	<u>0.62</u> (0.00)	<u>0.74</u> (0.01)	<u>0.55</u> (0.01)	**321K**	$\leq$ **0.02**
BackFlip-2⋆	**0.87** (0.01)	**0.58** (0.00)	**0.78** (0.01)	**0.34** (0.00)	<u>965K</u>	$\leq$ **0.02**

† pLM embeddings ⋆ One-hot sequence encoding

Table 2: BackFlip-2 outperforms DynaProt and is competitive with large ensemble generation models. We report the performance on the task of predicting per-residue covariance and pairwise residue couplings on the ATLAS test set (AlphaFlow split). All metrics are reported as a median over test set proteins, as in AlphaFlow, with deviation computed by bootstrapping MD trajectories 1000 times. For ensemble generation models, the total inference time required to generate 200 conformations (as in [25]) for a protein of 300 residues is reported.

Method	Variance contrib. of		RMSF		DCCM		# Params	Time [s]
	RMWD (↓)	KL (↓)	r (↑)	MAE (↓)	r (↑)	MAE (↓)		
MD (Gr. Tr.)	0.78 (0.01)	0.57 (0.00)	0.91 (0.00)	0.34 (0.00)	0.85 (0.00)	0.14 (0.00)	-	$\mathcal{O}(10{,}000)$
NMA (ANM)	1.46 (0.02)	4.55 (0.07)	0.79 (0.00)	1.03 (0.02)	0.78 (0.00)	0.20 (0.0)	31M	~ 2
BioEmu	1.99 (0.03)	2.66 (0.05)	0.84 (0.00)	1.40 (0.05)	0.80 (0.00)	0.19 (0.00)	31M	~ 380
AFMD-T	**0.84** (0.01)	**0.60** (0.00)	**0.93** (0.00)	**0.25** (0.00)	**0.89** (0.00)	**0.12** (0.00)	95M	~ 8200
BBFlow	0.93 (0.01)	<u>0.65</u> (0.01)	**0.93** (0.00)	0.42 (0.00)	<u>0.86</u> (0.00)	<u>0.16</u> (0.00)	18.2M	~ 160
DynaProt	1.18	0.91	0.87	-	0.66	-	<u>955K / 2.86M⋆</u>	**0.02**
BackFlip-2	<u>0.88</u> (0.00)	0.67 (0.01)	0.89 (0.00)	<u>0.35</u> (0.00)	0.80 (0.00)	<u>0.16</u> (0.00)	**965K**	$\leq$ **0.02**

⋆ Separately trained models for per-residue covariance / pairwise couplings.

the predicted per-residue covariances (RMWD Var. and KL Var.), yields more accurate reconstructed RMSF, and most notably captures dynamic residue couplings far better than DynaProt while having almost 3 times less trainable parameters (Tab. 2). The improvement in metrics is well above the finite-sampling noise of the reference MD trajectories (see App. A.10). At the same time, BackFlip-2 remains competitive with the much larger ensemble generation models AlphaFlow and BBFlow on per-residue covariance metrics, but underperforms on predicting residue couplings. However, our model is over 5 orders of magnitude faster than AlphaFlow and 4 orders of magnitude faster than BBFlow. Examples of predicted pairwise residue couplings are illustrated in Fig. 2.

4.2 Evaluation of anisotropy of the predicted covariances

For each predicted covariance matrix $\widehat{\Sigma}^{(ii)}$, the eigenvectors define the principal directions of fluctuation and the corresponding eigenvalues determine the magnitude of fluctuations along these directions. Since the directionality of residue motions is of particular interest for describing functional protein dynamics (see Sec. 1), we explicitly evaluate the shape of the predicted covariances. To this end, we construct ellipsoids from the eigendecomposition of the predicted and MD covariance matrices and quantify their volume overlap using IoU and Dice metrics, as described in App. A.2. We find that

Table 3: Evaluation of covariance directionality on the ATLAS test set, measured by volume overlap of the induced ellipsoids. IoU: Intersection over Union. Best values in bold, second-best underlined.

Model	Equiv.	Ensemble gen.	IoU (↑)	Dice (↑)
Ours	✓	×	**0.49**	**0.64**
Ours	×	×	0.42	0.58
BioEmu	-	✓	0.25	0.38
AFMD-T	-	✓	<u>0.48</u>	<u>0.63</u>
BBFlow	-	✓	0.47	0.62

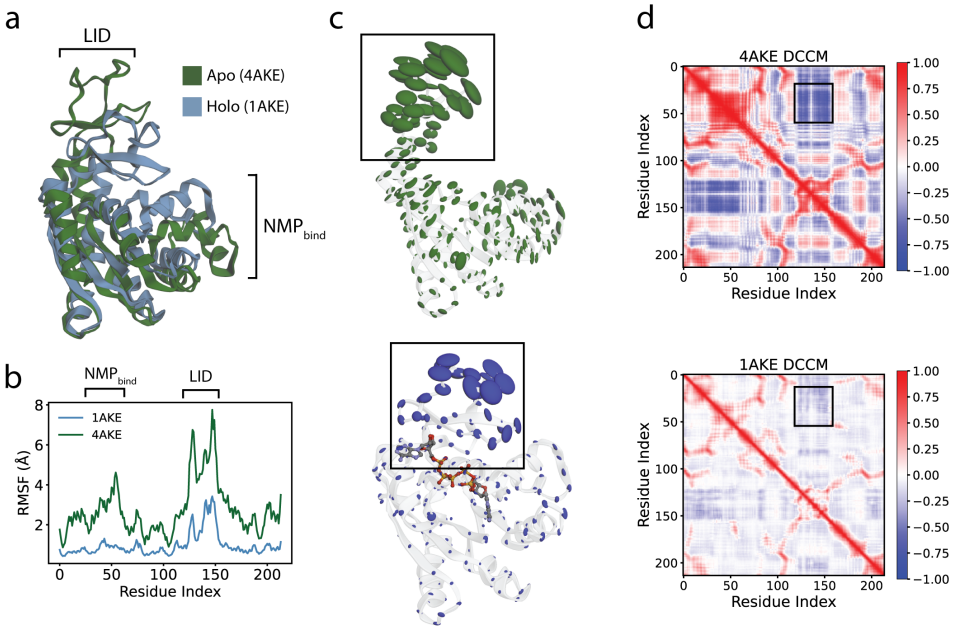


Figure 3: BackFlip-2 predictions for Adenylate Kinase (AdK). (a) The apo (4AKE) → holo (1AKE) major conformational transition (∼8 Å) closes the LID and NMPbind domains over the active site. (b) Both domains are predicted to be more flexible in the apo state, with (c) highly anisotropic motions along the closure-opening axis and (d) strongly negative pairwise couplings between LID and NMPbind residues, capturing their concurrent movement towards each other.

enforcing $\mathrm{SE}(3)$ equivariance in the covariance prediction head substantially improves ellipsoid volume overlap with MD compared to an $\mathrm{SE}(3)$-invariant model variant (Tab. 3). Crucially, our proposed equivariant model recovers anisotropic ellipsoids that match MD (Fig. 1), whereas the invariant parametrization yields nearly isotropic predicted covariances with spherical shape. Notably, our model also slightly outperforms expensive ensemble generative models AFMD-T and BBFlow at capturing the anisotropy of fluctuations.

4.3 Evaluation on functional protein motions and NMR ensembles

We predicted covariance ellipsoids and pairwise couplings for adenylate kinase (AdK), which closes its LID and NMPbind domains over the active site in the ~ 8 Å apo (4AKE) to holo (1AKE) transition. In the apo state the model predicts substantially higher flexibility, especially in the LID and NMPbind domains, with the highly anisotropic predicted movement oriented toward the active site, and substantially stronger anti-correlated NMPbind-LID coupling (i.e. movement towards each other) compared with the holo state (Fig. 3). This state-specific differentiation is consistent with the established mechanism of domain closure [41, 42, 16] and rigidification of AdK [42, 43], while the persistence of the negative LID–NMPbind coupling in the compact, rigidified holo state suggests that the apo-holo interconversion proceeds through concerted movement of the two domains. We provide additional results for dihydrofolate reductase (DHFR), where the model predicts functionally relevant catalytic Met20 loop behavior despite a ~ 0.8 Å RMSD between open and closed conformations, in App. A.5. Furthermore, without any training on NMR data, the model very closely reproduces the structural heterogeneity and coupling patterns of NMR ensembles and outperforms direct flexibility-prediction baselines, further suggesting transfer beyond the MD settings (App. A.6 and Tab. A.1).

4.4 Extension to the mdCATH dataset

We additionally evaluate ATLAS-trained BackFlip-2 on the mdCATH dataset to assess generalization across simulation settings, as in FlexPert [30]. We find that the ATLAS-trained model transfers well to the longer 500 ns timescale and higher-temperature mdCATH setting (see Sec. 1.1), particularly on RMSF and pairwise residue couplings, whereas FlexPert and the predecessor BackFlip show drop

in performance compared to their results on ATLAS (Tab. A.2). In addition, we train BackFlip-2 on a joint ATLAS/mdCATH dataset (App. A.3). Joint training yields only modest gains on mdCATH, with slight improvements for per-residue covariance predictions. If evaluated on the ATLAS test set, incorporating mdCATH data into training slightly degrades performance (Tab. A.3). Overall, the ATLAS-only model remains robust and readily applicable across different simulation regimes.

4.5 Ablations

Ablating model components We ablate the use of $\mathrm{SE}(3)$ equivariance and the SPSD parametrization proposed in Sec. 3.2 and report the results in Tab. A.4. We find that replacing equivariance by invariance in the per-residue covariance head leads to a substantial decrease in the performance, as can also be seen in Fig. 1. Further, the simple SPDS parametrization proposed in Eq. 9 performs slightly better than the Cholesky factorization approach introduced in DynaProt [33] (entry b).

Learning curve To investigate how performance of the proposed model scales with the amount of training data, we report performance on the ATLAS test set as a function of the training dataset size (Fig. A.2). With as little as 10% of the training data, BackFlip-2 matches the RMSF prediction performance of Pegasus, FlexPert and the predecessor BackFlip, and reaches the performance of DynaProt with approximately 50% of the data. For per-residue covariances, our model outperforms DynaProt using only 10% of the training data, showcasing the importance of $\mathrm{SE}(3)$ equivariance as inductive bias for this task.

Noised or AlphaFold-predicted structures as input We further test sensitivity of the model to the quality of the input backbone by perturbing the equilibrium structures of ATLAS proteins with Gaussian noise and by replacing them with AlphaFold2 (AF2) predictions (Tab. A.5). Performance is only slightly affected in both settings, even though the AF2 structures deviate from the equilibrium states by a median global RMSD of ~ 2.6 Å. This suggests the model is readily applicable to predicted or moderately perturbed backbones.

4.6 Limitations

Compared to the large ensemble generative models trained on the ATLAS dataset, BackFlip-2 is slightly less accurate, however, it is substantially faster at inference and outperforms the more comparable direct flexibility prediction baselines. As described in Sec. 1, the approach is fundamentally limited to predicting anisotropic motions and couplings occuring on the 300ns and 500ns MD timescale covered in the ATLAS and mdCATH datasets. It thus primarily captures sub-microsecond transitions and local degrees of freedom and cannot be expected to accurately describe slow, large-scale conformational changes, similar to all considered baselines except for BioEmu. This is because (i) the training data does not contain longer MD trajectories and (ii) the dynamical descriptors from Sec. 2 are suitable for fluctuations that are approximately Gaussian, as we also discuss in App. A.9, which is not the case for typical millisecond transitions between distant conformations.

5 Conclusions

We introduced BackFlip-2, an SE(3)-equivariant neural network for predicting directional flexibility and pairwise residue couplings from the equilibrium structure of a protein. We demonstrated that our method achieves state-of-the-art performance in terms of accuracy and inference time among flexibility prediction models on the established ATLAS benchmark dataset. Our method is either slightly less accurate or on-par with substantially larger ensemble generation models while being orders of magnitude faster. In ablations we show that SE(3) equivariance is a well-suited inductive bias for learning anisotropic covariances, whereas invariant models tend to predict spherical covariances and hence underperform. Correctly predicting these directional degrees of freedom is especially important in practice since anisotropic residue motions describe functional protein rearrangements, including hinge motion, allosteric communication, and molecular recognition [13, 12, 44]. We therefore expect our model to serve as a fast and accurate tool for characterizing anisotropic functional dynamics across large de novo or natural protein datasets such as the AlphaFoldDB [45], especially in cases where ensembles are not necessarily required - for instance to identify flexible functional sites, map residue coupling patterns, or screen for putative allosteric networks across protein families.

Acknowledgments This study was funded by Vetenskapsrådet, Grant no. 2021–03979, and the Knut and Alice Wallenberg Foundation, Grant no. 2022.0032. The computations and data handling were enabled through grant agreement no. 2021-29 and its continuation 2026-197 awarded by the supercomputing resource Berzelius, provided by the National Supercomputer (NAISS) and the Knut and Alice Wallenberg (KAW) Foundation.

References

[1] Donald Voet and Judith G Voet. *Biochemistry*. John Wiley & Sons, 2010.

[2] Alena Khmelinskaia, Neville P Bethel, Farzad Fatehi, Bhoomika Basu Mallik, Aleksandar Antanasijevic, Andrew J Borst, Szu-Hsueh Lai, Ho Yeung Chim, Jing Yang 'John' Wang, Marcos C Miranda, et al. Local structural flexibility drives oligomorphism in computationally designed protein assemblies. *Nature Structural & Molecular Biology*, 32(6):1050–1060, 2025.

[3] Steven D Schwartz. Protein dynamics and enzymatic catalysis. *The Journal of Physical Chemistry B*, 127(12):2649–2660, 2023.

[4] Kwangho Nam and Magnus Wolf-Watz. Protein dynamics: The future is bright and complicated! *Structural Dynamics*, 10(1), 2023.

[5] Siyuan Du, Rachael C Kretsch, Jacob Parres-Gold, Elisa Pieri, Vinícius Wilian D Cruzeiro, Mingning Zhu, Margaux M Pinney, Filip Yabukarski, Jason P Schwans, Todd J Martínez, et al. Conformational ensembles reveal the origins of serine protease catalysis. *Science*, 387(6735): eado5068, 2025.

[6] Wei Huang, Gessica M Adornato, Maggie Horst, Turki M Alturaifi, Kaipeng Hou, Peng Liu, William F DeGrado, and Yang Yang. De novo design, directed evolution and computational study of heme-binding helical bundle protein catalysts for biocatalytic enantioselective ge–h insertion. *Journal of the American Chemical Society*, 147(44):40869–40878, 2025.

[7] Audrey Tousignant and Joelle N Pelletier. Protein motions promote catalysis. *Chemistry & biology*, 11(8):1037–1042, 2004.

[8] Zeynep Kurkcuoglu, Ahmet Bakan, Duygu Kocaman, Ivet Bahar, and Pemra Doruker. Coupling between catalytic loop motions and enzyme global dynamics. 2012.

[9] Francesca Spyrakis, Axel BidonChanal, Xavier Barril, and F Javier Luque. Protein flexibility and ligand recognition: challenges for molecular modeling. *Current topics in medicinal chemistry*, 11(2):192–210, 2011.

[10] Peter Csermely, Robin Palotai, and Ruth Nussinov. Induced fit, conformational selection and independent dynamic segments: an extended view of binding events. *Trends in biochemical sciences*, 35(10):539–546, 2010.

[11] Krystle Reiss, Victor S Batista, and J Patrick Loria. Allosteric control of enzyme activity: from ancient origins to recent gene-editing technologies. *Biochemistry*, 59(18):1711–1712, 2020.

[12] Kaare Teilum, Johan G Olsen, and Birthe B Kragelund. Functional aspects of protein flexibility. *Cellular and Molecular Life Sciences*, 66(14):2231–2247, 2009.

[13] Lei Zhou and Qinglian Liu. Aligning experimental and theoretical anisotropic b-factors: water models, normal-mode analysis methods, and metrics. *The Journal of Physical Chemistry B*, 118 (15):4069–4079, 2014.

[14] Edward C Twomey and Alexander I Sobolevsky. Structural mechanisms of gating in ionotropic glutamate receptors. *Biochemistry*, 57(3):267–276, 2018.

[15] Nicole Stéphanie Galenkamp, Sarah Zernia, Yulan B Van Oppen, Marco van den Noort, Andreas Milias-Argeitis, and Giovanni Maglia. Allostery can convert binding free energies into concerted domain motions in enzymes. *Nature Communications*, 15(1):10109, 2024.

[16] Jie Ping, Pei Hao, Yi-Xue Li, and Jing-Fang Wang. Molecular dynamics studies on the conformational transitions of adenylate kinase: a computational evidence for the conformational selection mechanism. *BioMed Research International*, 2013(1):628536, 2013.

[17] Bernard Brooks and Martin Karplus. Harmonic dynamics of proteins: normal modes and fluctuations in bovine pancreatic trypsin inhibitor. *Proceedings of the National Academy of Sciences*, 80(21):6571–6575, 1983.

[18] Konrad Hinsen. Normal mode theory and harmonic potential approximations. In *Normal Mode Analysis*, pages 25–40. Chapman and Hall/CRC, 2005.

[19] Steven Hayward and Bert L De Groot. Normal modes and essential dynamics. In *Molecular Modeling of Proteins*, pages 89–106. Springer, 2008.

[20] Willem Diepeveen, Carlos Esteve-Yagüe, Jan Lellmann, Ozan Öktem, and Carola-Bibiane Schönlieb. Riemannian geometry for efficient analysis of protein dynamics data. *Proceedings of the National Academy of Sciences*, 121(33):e2318951121, 2024.

[21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851, 2020.

[22] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *International Conference on Learning Representations*, 2023.

[23] M. S. Albergo and E. Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *The Eleventh International Conference on Learning Representations*, 2022.

[24] Sarah Lewis, Tim Hempel, José Jiménez-Luna, Michael Gastegger, Yu Xie, Andrew YK Foong, Victor García Satorras, Osama Abdin, Bastiaan S Veeling, Iryna Zaporozhets, et al. Scalable emulation of protein equilibrium ensembles with generative deep learning. *Science*, 389(6761): eadv9817, 2025.

[25] Bowen Jing, Bonnie Berger, and Tommi Jaakkola. Alphafold meets flow matching for generating protein ensembles. *arXiv preprint arXiv:2402.04845*, 2024.

[26] Nicolas Wolf, Leif Seute, Vsevolod Viliuga, Simon Wagner, Jan Stühmer, and Frauke Gräter. Learning conformational ensembles of proteins based on backbone geometry. *arXiv preprint arXiv:2503.05738*, 2025.

[27] Yan Wang, Lihao Wang, Yuning Shen, Yiqun Wang, Huizhuo Yuan, Yue Wu, and Quanquan Gu. Protein conformation generation via force-guided se (3) diffusion models. In *Forty-first International Conference on Machine Learning*, 2024.

[28] Bowen Jing, Hannes Stark, Tommi Jaakkola, and Bonnie Berger. Generative modeling of molecular dynamics trajectories. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[29] Yann Vander Meersche, Gabriel Cretin, Aria Gheeraert, Jean-Christophe Gelly, and Tatiana Galochkina. Atlas: protein flexibility description from atomistic molecular dynamics simulations. *Nucleic acids research*, 52(D1):D384–D392, 2024.

[30] Petr Kouba, Joan Planas-Iglesias, Jiri Damborsky, Jiri Sedlar, Stanislav Mazurenko, and Josef Sivic. Learning to engineer protein flexibility. *arXiv preprint arXiv:2412.18275*, 2024.

[31] Yann Vander Meersche, Gabriel Duval, Gabriel Cretin, Aria Gheeraert, Jean-christophe Gelly, and Tatiana Galochkina. Pegasus: Prediction of md-derived protein flexibility from sequence. *Protein Science*, 34(8):e70221, 2025.

[32] Vsevolod Viliuga, Leif Seute, Nicolas Wolf, Simon Wagner, Arne Elofsson, Jan Stühmer, and Frauke Gräter. Flexibility-conditioned protein structure design with flow matching. *arXiv preprint arXiv:2508.18211*, 2025.

[33] Mihir Bafna, Bowen Jing, and Bonnie Berger. Learning residue level protein dynamics with multiscale gaussians. *ArXiv*, pages arXiv–2509, 2025.

[34] 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.

[35] Antonio Mirarchi, Toni Giorgino, and Gianni De Fabritiis. mdcath: A large-scale md dataset for data-driven computational biophysics. *Scientific Data*, 11(1):1299, 2024.

[36] Christine A Orengo, Frances MG Pearl, James E Bray, Annabel E Todd, AC Martin, L Lo Conte, and Janet M Thornton. The cath database provides insights into protein structure/function relationships. *Nucleic acids research*, 27(1):275–279, 1999.

[37] Hannah K. Wayment-Steele, Adedolapo Ojoawo, Renee Otten, Julia M. Apitz, Warintra Pitsawong, Marc Hömberger, Sergey Ovchinnikov, Lucy Colwell, and Dorothee Kern. Predicting multiple conformations via sequence clustering and alphafold2. *Nature*, 625:832–839, 2024.

[38] Markus Herdin, Nicolai Czink, Hüseyin Ozcelik, and Ernst Bonek. Correlation matrix distance, a meaningful measure for evaluation of non-stationary mimo channels. In *2005 IEEE 61st vehicular technology conference*, volume 1, pages 136–140. IEEE, 2005.

[39] Joseph L. Watson, David Juergens, Nathaniel R. Bennett, Brian L. Trippe, Jason Yim, Helen E. Eisenach, Woody Ahern, Andrew J. Borst, Robert J. Ragotte, Lukas F. Milles, Basile I. M. Wicky, Nikita Hanikel, Samuel J. Pellock, Alexis Courbet, William Sheffler, Jue Wang, Preetham Venkatesh, Isaac Sappington, Susana Vázquez Torres, Anna Lauko, Valentin De Bortoli, Emile Mathieu, Sergey Ovchinnikov, Regina Barzilay, Tommi S. Jaakkola, Frank DiMaio, Minkyung Baek, and David Baker. De novo design of protein structure and function with rfdiffusion. *Nature*, pages 1–3, 2023.

[40] Jason Yim, Andrew Campbell, Andrew Y. K. Foong, Michael Gastegger, José Jiménez-Luna, Sarah Lewis, Victor Garcia Satorras, Bastiaan S. Veeling, Regina Barzilay, Tommi Jaakkola, and Frank Noé. Fast protein backbone generation with se (3) flow matching. *arXiv preprint arXiv:2310.05297*, 2023.

[41] Christoph W Müller, Gerwald J Schlauderer, Jochen Reinstein, and Georg E Schulz. Adenylate kinase motions during catalysis: an energetic counterweight balancing substrate binding. *Structure*, 4(2):147–156, 1996.

[42] Michael A Sinev, Elena V Sineva, Varda Ittah, and Elisha Haas. Domain closure in adenylate kinase. *Biochemistry*, 35(20):6425–6437, 1996.

[43] Yury E Shapiro, Michael A Sinev, Elena V Sineva, Vitali Tugarinov, and Eva Meirovitch. Backbone dynamics of escherichia coli adenylate kinase at the extreme stages of the catalytic cycle studied by 15n nmr relaxation. *Biochemistry*, 39(22):6634–6644, 2000.

[44] Jingjing Guo and Huan-Xiang Zhou. Protein allostery and conformational dynamics. *Chemical reviews*, 116(11):6503–6515, 2016.

[45] Mihaly Varadi, Stephen Anyango, Mandar Deshpande, Sreenath Nair, Cindy Natassia, Galabina Yordanova, David Yuan, Oana Stroe, Gemma Wood, Agata Laydon, et al. Alphafold protein structure database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic acids research*, 50(D1):D439–D444, 2022.

[46] Michel Van Kempen, Stephanie S Kim, Charlotte Tumescheit, Milot Mirdita, Jeongjae Lee, Cameron LM Gilchrist, Johannes Söding, and Martin Steinegger. Fast and accurate protein structure search with foldseek. *Nature biotechnology*, 42(2):243–246, 2024.

[47] Michael R Sawaya and Joseph Kraut. Loop and subdomain movements in the mechanism of escherichia coli dihydrofolate reductase: crystallographic evidence. *Biochemistry*, 36(3): 586–603, 1997.

[48] Michael J Osborne, Jason Schnell, Stephen J Benkovic, H Jane Dyson, and Peter E Wright. Backbone dynamics in dihydrofolate reductase complexes: role of loop flexibility in the catalytic mechanism. *Biochemistry*, 40(33):9846–9859, 2001.

[49] Kresten Lindorff-Larsen, Stefano Piana, Ron O Dror, and David E Shaw. How fast-folding proteins fold. *Science*, 334(6055):517–520, 2011.

A Appendix

A.1 Comparison to Cholesky parametrization

DynaProt [33] parametrizes the covariance head via a Cholesky factorization, predicting only the six elements of a lower-triangular factor L with a softplus-constrained positive diagonal, and reconstructing the covariance as $LL^\top$. The softplus constraint ensures that all eigenvalues of $LL^\top$ are strictly positive, enforcing SPD by construction.

Our $AA^\top$ parametrization instead allows $A \in \mathbb{R}^{3\times 3}$ to be unconstrained, so the predicted covariance can be rank-deficient. We argue this is the correct choice for two reasons. First, it is physically well-motivated: a zero eigenvalue in some direction means the residue fluctuates with zero amplitude along that direction, i.e. it is perfectly rigid in that direction – a valid configuration for buried or geometrically constrained residues. Forcing strict positive definiteness via a softplus constraint introduces a systematic bias against such predictions, even when they are correct. Second, strict positive definiteness is not required for training: where the matrix logarithm must be evaluated, a small εI shift is sufficient. For any SPSD matrix Σ, the regularized matrix $\Sigma_\varepsilon = \Sigma + \varepsilon I$ with $\varepsilon > 0$ is SPD, since

$$\begin{aligned} v^\top \Sigma_\varepsilon \, v &= v^\top (\Sigma + \varepsilon I)\, v \\ &= v^\top \Sigma \, v + \varepsilon \, v^\top I \, v \\ &= v^\top \Sigma \, v + \varepsilon \, \|v\|^2 \\ &\geq 0 + \varepsilon \, \|v\|^2 \\ &= \varepsilon \, \|v\|^2 > 0, \end{aligned} \tag{16}$$

where the inequality uses that Σ is SPSD, i.e. $v^\top \Sigma \, v \geq 0$ for all v. Since $v^\top \Sigma_\varepsilon \, v > 0$ for all $v \neq 0$, the matrix Σ_ε is SPD.

A.2 Metrics for evaluating per-residue covariances

RMWD Variance RMWD calculates the 2-Wasserstein distance between the predicted and reference distributions under the assumption that both distributions are Gaussian. Since our model predicts only per-residue covariance matrices and not mean coordinates, we center both distributions at the same equilibrium structure, analogous to DynaProt [33], so that the mean contribution vanishes. We therefore report only the variance term:

$$\mathrm{RMWD}_{\mathrm{var}}\left(\widehat{\Sigma}, \Sigma\right) = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \mathrm{Tr}\left(\widehat{\Sigma}^{(ii)} + \Sigma^{(ii)} - 2\left(\widehat{\Sigma}^{(ii)}\Sigma^{(ii)}\right)^{1/2}\right)} \tag{17}$$

Symmetric KL Divergence Variance The symmetrized KL variance measures the divergence between the predicted and reference Gaussian distributions induced by their per-residue covariance blocks. Analogous to RMWD definition, we consider only the variance term. We compute the symmetric KL divergence variance by averaging the forward and reverse KL terms:

$$\mathrm{KL}_{\mathrm{symvar}}\left(\widehat{\Sigma}, \Sigma\right) = \frac{1}{4N}\sum_{i=1}^{N}\left[\mathrm{Tr}\left(\left(\Sigma^{(ii)}\right)^{-1}\widehat{\Sigma}^{(ii)}\right) + \mathrm{Tr}\left(\left(\widehat{\Sigma}^{(ii)}\right)^{-1}\Sigma^{(ii)}\right) - 2d\right], \tag{18}$$

where $d = 3$ for Cartesian per-residue covariance blocks.

Volume overlap of ellipsoids induced by the per-residue covariance Although RMWD Variance and Symmetric KL Variance measure discrepancies between the distributions induced by the predicted and ground-truth covariances, they summarize disagreement as scalar distributional distances. However, we are interested in evaluating whether the predicted covariances recover the anisotropic residue fluctuations and the orientation of their principal axes, which are observed in MD (Fig. 1). To explicitly assess this, we construct ellipsoids from the predicted and MD-derived per-residue covariance blocks and compare their volume overlap.

Concretely, for each per-residue covariance block $\Sigma^{(ii)} \in \mathbb{R}^{3\times3}$, we construct the centered covariance ellipsoid

$$\mathcal{E}\left(\Sigma^{(ii)}\right) = \left\{ x \in \mathbb{R}^3 : x^\top \left(\Sigma^{(ii)}\right)^{-1} x \leq 1 \right\}, \tag{19}$$

where x is a centered Cartesian displacement vector. To obtain an intuition for the geometric relation between this ellipsoid and the covariance matrix, we note that the principal axes and semi-axis lengths of the ellipsoid can be obtained from the eigendecomposition of the covariance for a given residue i:

$$\Sigma^{(ii)} = Q_i \,\mathrm{diag}\left(\lambda_{i1}, \lambda_{i2}, \lambda_{i3}\right) Q_i^\top, \qquad r_{ij} = \sqrt{\lambda_{ij}}. \tag{20}$$

The columns of Q_i define the ellipsoid principal directions, and r_{ij} gives the semi-axis length along the j-th principal direction.

For a unit direction $u \in \mathbb{S}^2$, the radial extent of $\mathcal{E}(\Sigma^{(ii)})$ is

$$r_{\Sigma^{(ii)}}(u) = \left(u^\top \left(\Sigma^{(ii)}\right)^{-1} u \right)^{-1/2}. \tag{21}$$

For predicted and ground-truth covariance blocks $\widehat{\Sigma}^{(ii)}$ and $\Sigma^{(ii)}$, we then define the radial extents of their intersection $r_i^{\cap}$ and union $r_i^{\cup}$ as

$$r_i^{\cap}(u) = \min\left(r_{\widehat{\Sigma}^{(ii)}}(u), r_{\Sigma^{(ii)}}(u)\right), \quad r_i^{\cup}(u) = \max\left(r_{\widehat{\Sigma}^{(ii)}}(u), r_{\Sigma^{(ii)}}(u)\right). \tag{22}$$

We then compute **IoU** (Intersection over Union) as the volume ratios of intersection and union of the ellipsoids,

$$\mathrm{IoU}_i = \frac{\int_{\mathbb{S}^2} \left[r_i^{\cap}(u)\right]^3 \, d\Omega}{\int_{\mathbb{S}^2} \left[r_i^{\cup}(u)\right]^3 \, d\Omega}, \tag{23}$$

and the **Dice** overlap as

$$\mathrm{Dice}_i = \frac{2\int_{\mathbb{S}^2} \left[r_i^{\cap}(u)\right]^3 \, d\Omega}{\int_{\mathbb{S}^2} r_{\widehat{\Sigma}^{(ii)}}(u)^3 \, d\Omega + \int_{\mathbb{S}^2} r_{\Sigma^{(ii)}}(u)^3 \, d\Omega}. \tag{24}$$

IoU and Dice measure the overlap between the predicted and reference covariance-induced ellipsoids, hence measuring the similarity both in terms of magnitude but also in terms of shape, capturing differences in anisotropy.

A.3 Joint ATLAS/mdCATH dataset construction

We construct a joint ATLAS/mdCATH dataset by assigning mdCATH domains to the existing AlphaFlow [25] ATLAS train, validation, and test splits based on structural similarity. First, we cluster proteins from ATLAS and mdCATH using FoldSeek [46] in TMalign mode with a TM-score threshold of 0.5. We then use the original AlphaFlow ATLAS split assignment as a fixed reference. Each mdCATH domain is assigned to the train, validation, or test split if it clusters with at least one ATLAS protein from the corresponding split. Domains that do not cluster with any ATLAS protein are randomly assigned to the train, validation, and test splits in an 80/10/10 ratio. If an mdCATH domain is associated with ATLAS proteins from more than one split, we discard it to avoid structural overlap across splits. This procedure yields a final split that is approximately 80/10/10, with 5673 proteins in the training set, 531 in the validation set, and 508 in the test set. The joint ATLAS/mdCATH datasplit can be obtained at `https://github.com/graeter-group/backflip`.

A.4 Details on flexibility prediction baselines

Several deep learning (DL) models have been proposed to predict isotropic flexibility of protein backbones directly, most commonly quantified as per-residue Root Mean Square Fluctuation (RMSF), which measures the average positional deviation from the equilibrium state across a conformational ensemble. FlexPert [30] fine-tunes a pretrained protein Language Model (pLM) and fuses its embeddings with external structural features with a learnable CNN module. Pegasus [31] predicts RMSF from embeddings of 4 different pretrained pLMs with a CNN and does not require structure as input. BackFlip [32] instead proposes to infer RMSF with an SE(3)-invariant graph neural network using only protein backbone geometry. All these models are trained and evaluated on ATLAS.

A.5 Prediction of flexibility and pairwise couplings in DHFR

In addition to the adenylate kinase experiment (Sec. 4.3 and Fig. 3), we evaluate the model on *E. coli* dihydrofolate reductase (DHFR), comparing the predictions made for two experimentally resolved structures with the same set of bound ligands (folate and $NADP^+$) but differing in the conformation of the catalytic Met20 loop: open (1RA2) and closed (1RX2). The open conformation has been proposed to act as an intermediate along the closed-occluded interconversion that precedes product release [47]. Despite a global RMSD of only 0.8 Å between the two states, the model predicts markedly higher Met20 loop flexibility in the open state, with anisotropy oriented along the closure direction (Fig. A.1). NMR relaxation experiments have reported strongly reduced loop flexibility in the closed complex, attributed to tighter packing and engagement of the Met20 loop [48]. In the closed state, the model further predicts more positive couplings between the Met20 loop and residues lining the ligand-binding site, i.e. more strongly aligned motion, as could be expected if the loop moves coherently with the binding site residues.

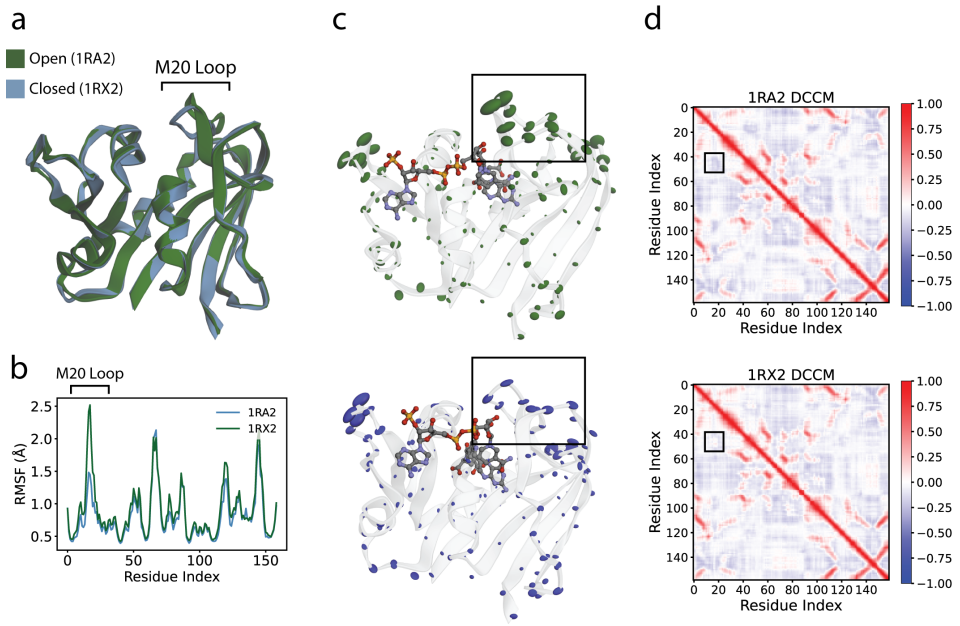


Figure A.1: BackFlip-2 predictions for Dihydrofolate Reductase (DHFR). (a) DHFR undergoes a transition between the closed state (1RX2) to the occluded state proposed to occur through the open (1RA2) state, which only differ by 0.8 Å in the catalytic Met20 loop. (b) BackFlip-2 predicts higher Met20-loop flexibility in the open state, (c) with the predicted anisotropic motion consistent with an opening-closing coordinate. (d) In the closed state, couplings between the Met20 loop and residues lining the ligand-binding site become more positive, indicating that the loop moves in concert with the binding site.

A.6 Performance on recapitulating the structural uncertainty of NMR ensembles

We further evaluated the model on 2000 randomly selected in-solution NMR ensembles from the PDB, each with at least 10 deposited conformations. We note that deposited NMR conformations cannot be considered as a statistic, thermodynamics ensemble, they rather represent the structural uncertainty consistent with the experimental restraints, averaged over heterogeneous timescales. We therefore consider conformational variability deposited as NMR ensembles as an independent, experimentally grounded proxy for local structural heterogeneity rather than the ground truth for equilibrium populations. Despite never having been trained on NMR data, the model very closely reproduces both RMSF and the pairwise couplings (Tab. A.1), outperforming direct flexibility-prediction baselines.

Table A.1: Performance of BackFlip-2 and baselines at recapitulating structural heterogeneity of 2000 randomly selected solution NMR ensembles from the PDB. We do not report FlexPert [30] performance here because inference crashes in the preprocessing pipeline of the released code. DynaProt [33] is omitted since the code was not made publicly available.

Method	RMWD Var. (↓)	KL Var. (↓)	RMSF r (↑)	RMSF MAE (↓)	DCCM r (↑)	DCCM MAE (↓)
Pegasus	-	-	0.83	0.93	-	-
BackFlip	-	-	0.90	0.70	-	-
BackFlip-2	1.2	1.9	**0.91**	**0.62**	0.80	0.18

A.7 Effect of joint training

Table A.2: Performance on 508 mdCATH proteins at 320 K for models trained either on ATLAS alone or jointly on ATLAS/mdCATH, compared with FlexPert (not trained on mdCATH). Metrics are reported as median over all proteins in the mdCATH dataset. The description on construction of the joint ATLAS/mdCATH dataset can be found in App. A.3. Note that we cannot compare to DynaProt [33] on the same dataset because the source code is not available.

Method	RMWD Var. (↓)	KL Var. (↓)	RMSF r (↑)	RMSF MAE (↓)	DCCM r (↑)	DCCM MAE (↓)
FlexPert	-	-	0.70	0.98	-	-
BackFlip	-	-	0.77	0.84	-	-
BackFlip-2-ATLAS	1.63	2.57	0.85	0.69	**0.81**	0.15
BackFlip-2-joint	**1.51**	**1.72**	**0.88**	**0.65**	**0.81**	**0.14**

Table A.3: Performance on 82 ATLAS test proteins (AlphaFlow split) for models trained either on ATLAS alone or jointly on ATLAS/mdCATH. Metrics are reported as median over all proteins in the mdCATH dataset. The description on construction of the joint ATLAS/mdCATH dataset can be found in App. A.3

Method	RMWD Var. (↓)	KL Var. (↓)	RMSF r (↑)	RMSF MAE (↓)	DCCM r (↑)	DCCM MAE (↓)
BackFlip-2-ATLAS	**0.88**	**0.67**	0.89	**0.35**	**0.80**	**0.16**
BackFlip-2-joint	1.04	0.82	**0.90**	**0.35**	0.78	0.17

A.8 Ablation experiments

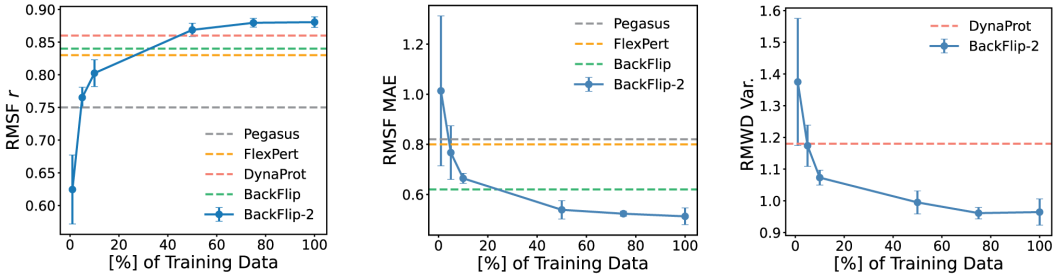


Figure A.2: Learning curves on the ATLAS test set. Our model outperforms flexibility prediction baselines using only a fraction of the training data. We report RMSF correlation and MAE, and RMWD of per-residue covariances as a function of the training dataset size. For our model, each point reports the mean and standard deviation over 5 training runs on the ATLAS dataset.

Table A.4: Ablation of BackFlip-2. We focus on the equivariance of the per-residue covariance output and the Cholesky factorization from DynaProt [33] compared to the proposed simpler parametrization (Eq. 9). All metrics are reported as medians over all proteins, analogous to Tab. 2. Best value is highlighted in bold, second-best is underlined.

Model	Equiv.	Per-res Cholesky	RMWD (↓)	Sym. KL (↓)	RMSF r (↑)	RMSF MAE (↓)	DCCM r (↑)	DCCM MAE (↓)
Ours	✓	×	**0.88**	**0.67**	<u>0.89</u>	<u>0.35</u>	**0.80**	**0.16**
a	×	×	1.05	1.06	0.88	0.38	0.71	0.20
b	✓	✓	<u>0.93</u>	<u>0.69</u>	**0.90**	<u>0.35</u>	<u>0.78</u>	<u>0.17</u>
c	×	✓	1.05	1.08	0.88	**0.33**	0.74	0.19

Table A.5: Performance of BackFlip-2 when either distorted equilibrium or AlphaFold2-predicted structures are provided as input. We add Gaussian noise with the indicated standard deviation to the backbone atoms of equilibrium structures, and rerun inference with the model for the ATLAS test set. Metrics are reported as in Tab. 2. **Bold** marks the best and <u>underline</u> the second-best value per column.

Method	RMWD Var. (↓)	KL Var. (↓)	RMSF r (↑)	RMSF MAE (↓)	DCCM r (↑)	DCCM MAE (↓)
BackFlip-2	**0.88**	**0.67**	**0.89**	**0.35**	**0.80**	**0.16**
BackFlip-2 +0.2 Å	<u>0.96</u>	<u>0.71</u>	<u>0.88</u>	**0.35**	<u>0.79</u>	**0.16**
BackFlip-2 +0.5 Å	0.99	0.82	**0.89**	0.40	**0.80**	<u>0.17</u>
BackFlip-2 + AF2	0.99	0.77	**0.89**	<u>0.36</u>	0.78	<u>0.17</u>

A.9 Limitations to learn the dynamical descriptors derived from long-timescale MD simulations

There are two main obstacles for extending our method to long- (millisecond to second) timescales. The first challenge is of practical nature. MD is expensive, hence existing millisecond-MD datasets are limited and cover only a few small proteins [49]. Approaches for modeling long-timescale dynamics hence rely on alternative tasks and are not trained to reproduce MD-derived properties. For example, BioEmu [24] is trained on structural ensembles derived from experiment and related proteins from the AlphaFold database. In principle, our method could also be trained on such a dataset. However, the second obstacle is more fundamental: the dynamical descriptors that our method learns, namely the per-residue covariance and pairwise couplings, rely on the assumption that the ground truth ensemble can be approximated as a Gaussian distribution (see Sec. 2, and Eq. 1). This Gaussianity assumption, in most cases, breaks down for large conformational changes and alternative folding states, which typically occur on long timescales. Hence the observables that the model would learn could lose their meaning. Even if they were predicted correctly (for example via coarse-grained representations), they would not be suitable to describe the biophysics of the system. We hence see ensemble generation models like BioEmu as more suitable for long timescales, and our method (of directly predicting the observables that are often of interest) as more suitable for shorter, nanoseconds-microsecond timescales.

A.10 Convergence of MD reference observables

We evaluated the convergence of the metrics reported in Tab. 2 on both ATLAS and mdCATH. For each ATLAS test protein, we concatenated the three 100 ns trajectories and computed per-residue covariance matrices, RMSF, and DCCMs from the full 300 ns trajectory. We then randomly sampled five time windows for each length 20, 100, 200, and 250 ns to estimate the deviation between such truncated trajectories and the full 300ns trajectory. At each trajectory length, we report the full set of metrics relative to the complete 300 ns trajectory. We conducted the same computation with the mdCATH dataset using trajectory lengths of 20, 100, 200, 300, and 500 ns. We report the metrics in Fig. A.3.

For ATLAS, we find that the dynamical descriptors are effectively converged before reaching 300 ns. At 250 ns, the RMSF- and DCCM-related metrics are fully converged, and the RMWD variance error relative to the full trajectory is approximately 0.02. RMSF and DCCM, global properties of the system,

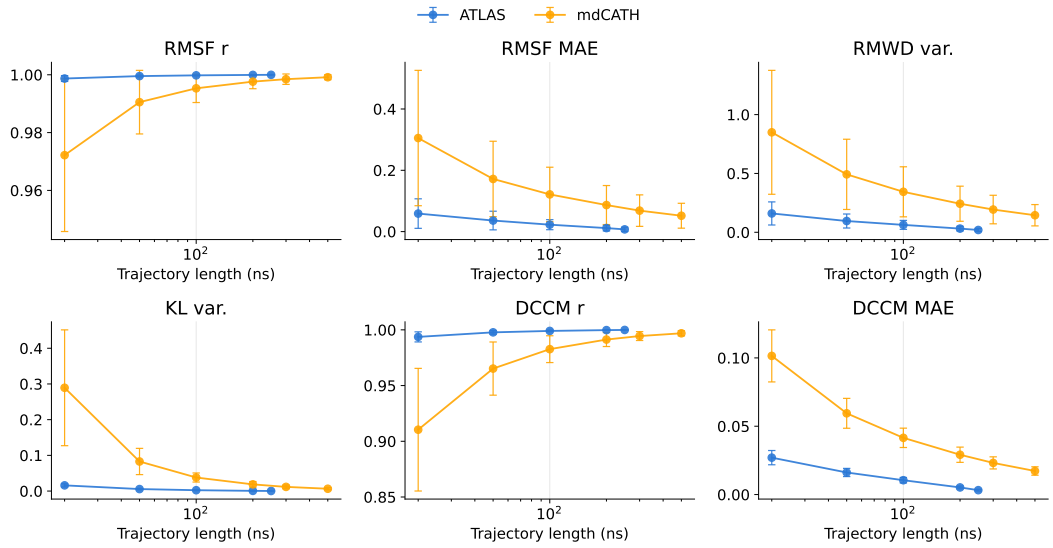


Figure A.3: Convergence of the evaluation metrics on ATLAS and mdCATH. Each point compares observables computed from a truncated trajectory at the indicated length against those derived from the full trajectory (300 ns for ATLAS, 500 ns for mdCATH).

converge faster than RMWD and the limit values are similar across the two datasets. Importantly, the ATLAS RMWD deviation decreases consistently towards zero with increasing trajectory length, indicating that it is convergent as well. Thus, the finite sampling noise is substantially smaller than the performance differences reported in Tab. 2. In case of mdCATH, RMWD converges towards a systematically higher value compared to ATLAS, possibly suggesting that the discrepancy between datasets cannot be explained solely by insufficient sampling and most likely, indeed, reflects differences in simulation protocol, including the higher temperature (320K) and integration timestep (4 fs) used for mdCATH.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes],

   Justification: As stated in the abstract and our main contributions, we propose an SE(3)-equivariant model for predicting per-residue anisotropic equivariant covariances and pairwise residue couplings (Sec. 3.2), and demonstrate that the proposed equivariance is well-suited to capture directional flexibility in proteins (Fig. 1, Tab. 4.2). Indeed, the model outperforms RMSF prediction baselines (Tab. 1), its most direct SE(3)-invariant baseline DynaProt, and is on-par with large ensemble generative models while being orders of magnitude faster (Tab. 2).

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have discussed the limitations of the model in a dedicated section (Sec. 4.6).

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: We provide the justification for using the equivariance when predicting anistotropic covariances in Sec. 2 and Eq. 3 and show the equivariance of our model in Sec. 3.2. We compare the Cholesky parametrization proposed in DynaProt [33] enforcing symmetric positive definiteness to our less constrained symmetric positive semidefinite (SPSD) construction of covariance predictions in Sec. A.1. We show that both equivariance and SPSD are fulfilled in Sec. 3.2.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: In Sec. 4 and App. A.2 we provide the full details on the metrics, baselines and publicly available dataset splits used to train our models. The newly introduced joint ATLAS/mdCATH dataset split is further explained in App. A.3. The source code, the model weights and evaluation scripts are available at `https://github.com/graeter-group/backflip`.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We train the model on the ATLAS dataset [29], which is publicly available. Our joint ATLAS/mdCATH dataset split is accessible along with the source code and model weights at `https://github.com/graeter-group/backflip`.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We use FlexPert [30] and AlphaFlow [25] dataset splits when training our model on ATLAS [29], which are both publicly available and also released along with the source code and model weights at `https://github.com/graeter-group/backflip`. We explain how the reported model checkpoint was obtained in Sec. 4. The hyperparameters of the model training are also accesible under configs in the repository linked above.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes],

Justification: For the learning curve ablation, we train 5 model initialized with different seeds and report mean and standard deviation in the Fig. A.2. For the rest, we report statistical significance via bootstrapping, as described in Table captions (Tab. 1 and 2).

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide time required to train a model and the hardware specification in Sec. 4. We also report the inference time for all models (Tab. 1 and Tab. 2).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: No human subjects or data with privacy concerns was used. Training was performed only on publicly available, academic datasets.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No],

    Justification: In the current paper version, due to the space restraints (9 pages max) we do not discuss the societal impact. Upon the release of the camera-ready version, this section will be included in the paper: *We consider the societal impact of this work as mostly positive, as understanding protein dynamics holds the promise to develop drugs and therapies against diseases, which outweighs potential risks.*

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper presents a method to describe structural fluctuations of proteins, which is commonly used by practitioners with standard methods such as MD or NMA, we thus assess the risk of misuse as very low.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The work builds on the Invariant Point Attention algorithm introduced in AlphaFold2 [34], which is publised under the Apache License 2.0, which permits creative derivative works. The source code is published under the MIT license and specifies the original license explicitly whenever applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We introduce a new joint ATLAS/mdCATH dataset split, which is described in App. A.2. These are accessible at `https://github.com/graeter-group/backflip`.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: No crowd sourcing or research with human subjects was conducted.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No institutional review board approval required.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLM was only used for editing and formatting the text.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.