Learning and Steering Protein Flexibility for Generative Protein Design

Anonymous authors
Paper under double-blind review

Abstract

Protein flexibility is a fundamental determinant of molecular recognition, allostery, and catalysis, yet predicting and engineering it at scale remains an open challenge. Existing approaches either rely on computationally prohibitive molecular dynamics simulations, resort to physics-based approximations that lack differentiability and miss sequence-dependent effects, or learn representations that are often tied to specific simulation protocols and can struggle to generalize across protein families and conditions. We introduce **FlexFlow**, a framework for designing proteins with prescribed per-residue root mean square fluctuation (RMSF) profiles via inference-time guidance on a frozen pretrained sequence–structure co-design model. At its core, FlexFlow is powered by **FlexFormer**, a differentiable predictor of per-residue backbone flexibility that achieves state-of-the-art accuracy and, unlike predictors that read only coordinates, responds to individual amino acid substitutions, tracking their measured effect on flexibility more faithfully than existing methods. Rather than retraining a conditional generative model, which may distort the structural prior learned from experimentally solved structures and sacrifices designability for flexibility, FlexFlow steers a pretrained joint sequence–structure co-design model at inference time using FlexFormer gradients, without any task-specific retraining. Each backbone is steered directly toward the target flexibility profile during generation, eliminating the post-hoc screening that existing methods require, yielding significant reduction in wall-clock time at substantially better structural quality.

1 Introduction

Protein function is inherently dynamic Henzler-Wildman & Kern (2007). Residue-level conformational fluctuations govern catalysis Benkovic et al. (2008); Miller & Benkovic (1998); Liao et al. (2018); Zinovjev et al. (2024), allosteric communication Yu & Koshland Jr (2001); Pillai et al. (2024), and molecular recognition including the formation of transient binding pockets relevant to drug discovery Glasscock et al. (2025); Pacesa et al. (2024); Hollingsworth et al. (2019). Yet most computational protein design pipelines optimize static structures, and deep learning-generated backbones tend to be highly rigid, an artefact of training on thermostable, crystallographically resolved proteins Watson et al. (2023); Zambaldi et al. (2024); Scandurra et al. (1998). Designing proteins with prescribed flexibility is therefore critical for engineering enzymes with optimized catalytic cycles Benkovic et al. (2008); Miller & Benkovic (1998), binders that can adapt to their targets Glasscock et al. (2025); Pacesa et al. (2024); Hollingsworth et al. (2019), and allosteric switches that respond to cellular signals Pillai et al. (2024); Yu & Koshland Jr (2001), yet incorporating such dynamic behavior into the design process remains an open challenge.

Flexibility prediction useful for design remains an open challenge because existing approaches each sacrifice one of the three properties that a useful predictor must have: accuracy, responsiveness to sequence perturbation, and differentiability with respect to backbone coordinates. Molecular dynamics simulations remain the gold standard for characterizing per-residue flexibility, but simulating a single protein to convergence requires days to weeks of specialized hardware time Shaw et al. (2010); Hollingsworth & Dror (2018), making

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

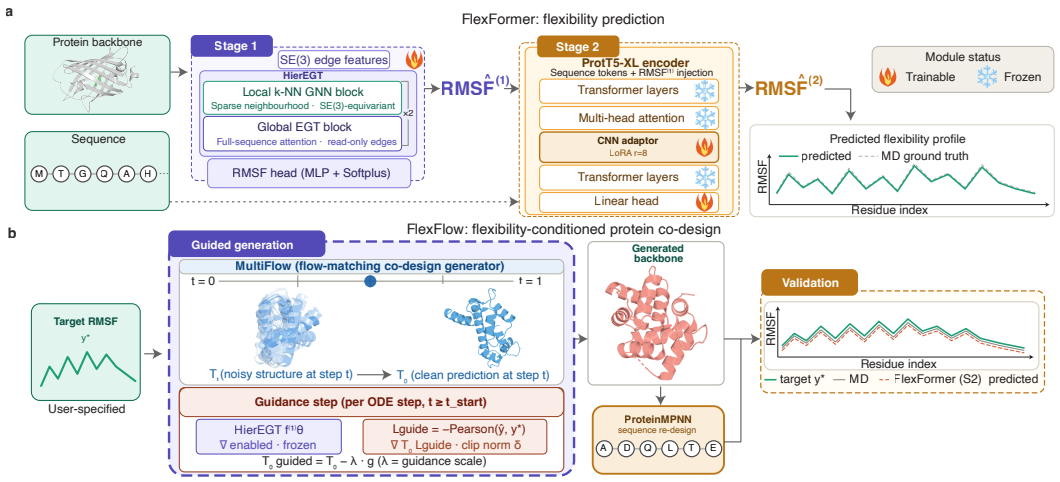


Figure 1: **Overview of FlexFormer and FlexFlow. (a) FlexFormer: flexibility prediction.** Stage 1 uses HierEGT, a hierarchical structure encoder, to predict per-residue RMSF from backbone Cα coordinates. Stage 2 refines this estimate using a frozen ProtT5-XL encoder with a trainable CNN adaptor injecting the Stage 1 RMSF signal alongside sequence embeddings. **(b) FlexFlow: flexibility-conditioned protein co-design.** A user-specified target flexibility profile y^* guides the MultiFlow ODE rollout via HierEGT gradients at each denoising step. The generated backbone is redesigned with ProteinMPNN and validated against y^* using MD simulation and FlexFormer predictions.

MD impractical as an in-the-loop signal for large-scale engineering. Physics-based elastic network models such as the Anisotropic Network Model Eyal et al. (2006) and related ENM approaches Bahar et al. (1997); Skjaerven et al. (2009) are computationally efficient but assume harmonic fluctuations around a single equilibrium conformation, cannot capture sequence-dependent or long-range effects, and are not differentiable with respect to backbone coordinates in a form suitable for gradient-based design. FlexPert-3D Kouba et al. (2024) takes a learned approach but relies on ProDy's ANM Eyal et al. (2006) as a feature extractor, making the pipeline not end-to-end differentiable with respect to backbone coordinates and precluding its use as a guidance signal during generative design. BackFlip Viliuga et al. (2025) achieves strong accuracy from structure alone, but as published it is a function of backbone coordinates only: substituting a residue moves no atom, so it returns an identical flexibility profile for every sequence variant and cannot rank the point substitutions that design applies. We identify this insensitivity to sequence as an under-recognized limitation of flexibility predictors: it is invisible to every benchmark in current use, since all of them hold sequence fixed, and to our knowledge this is the first work to measure it.

We address two coupled problems: predicting per-residue backbone flexibility accurately and responding correctly to sequence perturbation, and leveraging that predictor to engineer flexibility during generative protein design. Existing approaches fail at one or both. We introduce **FlexFlow**, a framework that solves both jointly, built around **FlexFormer**, a differentiable two-stage flexibility predictor that is accurate, perturbation-sensitive, and whose coordinate gradient is directly usable as a guidance signal during backbone generation. The design separates backbone geometry from sequence: a backbone-only Stage 1 supplies a per-residue rigidity estimate, which a pretrained protein language model refines from sequence through an additive readout. That readout is what makes both capabilities available at once, since the coordinate gradient reaches the output through Stage 1 alone while a substitution moves only the sequence term. Stage 2 adapts the Flexpert-3D architecture by coupling its frozen protein language model and convolutional adaptor to the Stage 1 flexibility estimate, which replaces the original elastic-network input channel. FlexFormer attains the best reported accuracy on ATLAS, retains its ranking on mdCATH, and is the only method that is simultaneously accurate and sensitive to the perturbation effect of a substitution.

Differentiability with respect to backbone coordinates is what connects FlexFormer to the generative design problem and makes FlexFlow possible. The central challenge in flexibility-conditioned design is that there is a fundamental tension in conditional generative training: optimizing for flexibility directly modifies the learned distribution over protein structures, trading away the structural diversity and physical validity that makes the generative model useful in the first place. Existing methods Viliuga et al. (2025) retrain the generative model to condition on flexibility, corrupting the structural prior learned from the PDB and forcing users to generate up to 100 candidate backbones per design and apply a separate screening step to compensate for the degraded output quality. FlexFlow resolves this tension entirely by leaving the pretrained joint sequence-structure co-design model Campbell et al. (2024) frozen and steering each backbone directly toward the target flexibility profile at inference time using gradients from FlexFormer's backbone encoder alone, without any retraining. This eliminates the need for candidate pools and post-hoc screening, and yields more than an order of magnitude reduction in wall-clock time at substantially better structural quality, validated by explicit molecular dynamics simulations on 40 generated backbones. Together, these contributions form a unified framework for flexibility-aware protein design, addressing a fundamental gap that static structure-based pipelines cannot fill. By making dynamic behavior a first-class design objective, this framework opens new avenues for engineering enzymes with optimized catalytic cycles, adaptive binders, and allosteric switches that respond to cellular signals.

2 Related Work

Protein flexibility estimation and prediction. Protein flexibility governs molecular recognition, allostery, and catalysis, and has been quantified through experimental techniques including crystallographic B-factors Sun et al. (2019), NMR order parameters, and hydrogen-deuterium exchange mass spectrometry. Computationally, molecular dynamics simulations provide residue-resolved flexibility profiles but remain prohibitively expensive for large-scale applications. Coarse-grained alternatives such as GNM Bahar et al. (1997) and ANM Eyal et al. (2006) approximate flexibility from contact topology alone, offering speed at the cost of ignoring sequence-dependent determinants. The availability of large-scale MD databases, including ATLAS Vander Meersche et al. (2024) and mdCATH Mirarchi et al. (2024), has enabled data-driven approaches: Flexpert Kouba et al. (2024) trains sequence-only and structure-aware models on ATLAS, with its structure variant injecting ANM estimates into a protein language model, while BackFlip Viliuga et al. (2025) leverages geometric algebra representations with MD-ensemble augmentation. AlphaFold2 pLDDT Jumper et al. (2021) and ESM pLDDT Lin et al. (2022) have also been repurposed as flexibility proxies, though they were designed for prediction confidence rather than physical dynamics. Despite these advances, the correlation these benchmarks rank is nearly exhausted, calibration goes unreported, and perturbation response is untested. We measure perturbation response at every position of all 139 ATLAS test proteins and all 500 mdCATH domains, and show that guidance from the predictor that leads on it produces more designable backbones than conditional training.

Protein language models and geometric deep learning. Pre-trained protein language models such as ESM-2 Lin et al. (2023), ESM-1v Meier et al. (2021), and ProtT5 Elnaggar et al. (2021) capture evolutionary constraints from large-scale sequence databases and have been applied to fitness prediction Meier et al. (2021); Hie et al. (2024), stability prediction Li & Luo (2025), and variant effect estimation Brandes et al. (2023). Parameter-efficient fine-tuning via LoRA Hu et al. (2022) and adapter modules Sledzieski et al. (2024) enables task-specific adaptation without full retraining. On the structural side, graph neural networks and edge-augmented transformers have become standard for protein modelling, with applications in inverse folding Dauparas et al. (2022) and fitness prediction Zhao et al. (2024); Luo et al. (2021). Our FlexFormer combines both paradigms: a hierarchical structure encoder (HierEGT) that separates local SE(3)-equivariant reasoning from global context aggregation, followed by a LoRA-adapted ProtT5 encoder that injects the learned structural signal alongside sequence embeddings.

Flexibility-conditioned protein design. Generative models for protein backbone design, including RFdiffusion Watson et al. (2023) and flow-matching approaches Yim et al. (2023);

Campbell et al. (2024), have enabled de novo structure generation but do not explicitly condition on dynamic properties. FliPS Viliuga et al. (2025) addresses this by training a conditional flow-matching model on ATLAS-derived flexibility profiles, but its dedicated conditional training limits generalization to unseen simulation conditions and protein families. Our FlexFlow instead performs inference-time guidance on a pretrained unconditional flow model (MultiFlow Campbell et al. (2024)), leveraging the full differentiability of HierEGT to steer backbone generation toward user-specified flexibility profiles without retraining, providing a continuous flexibility–designability trade-off.

3 FlexFormer: Backbone Flexibility Predictor

Designing proteins with prescribed flexibility requires a predictor that is simultaneously accurate, responsive to sequence perturbation, and differentiable with respect to backbone coordinates. Accuracy is needed to provide a reliable guidance signal; responsiveness is what lets the predictor rank the point substitutions design actually applies; and differentiability is what connects the predictor to the generative design loop. No existing method satisfies all three: ENM approaches (Eyal et al., 2006; Bahar et al., 1997) are not differentiable and miss sequence effects; FlexPert-3D (Kouba et al., 2024) is non-differentiable end-to-end due to its ANM Hessian featurizer; and BackFlip (Viliuga et al., 2025), as published, is a function of coordinates alone and returns the same profile for every sequence variant.

We introduce **FlexFormer**, a two-stage differentiable model for predicting per-residue protein flexibility (RMSF) from backbone structure and sequence, where given a protein of length N with sequence $S = (s_1, \ldots, s_N)$ and backbone C_α coordinates $\mathcal{X} \in \mathbb{R}^{N\times 3}$, we learn $f_\theta(S, \mathcal{X}) = \hat{y} \in \mathbb{R}^N$ with $\hat{y}_i$ predicting the per-residue RMSF of residue i averaged over three MD replicas from ATLAS (Vander Meersche et al., 2024). Stage 1 is a backbone-only predictor whose per-residue estimate is injected into Stage 2, a pretrained protein language model that refines it from sequence. Beyond profile accuracy, we ask whether the model captures the *perturbation effect* of a single substitution on flexibility, a capability a design screen requires and one that profile benchmarks cannot expose. The full model is end-to-end differentiable with respect to backbone coordinates, enabling its use as a guidance signal in our flexibility-conditioned design pipeline (Section 4).

3.1 Results

Table 1: **Per-residue RMSF prediction** ATLAS: 139 test proteins, official topology split of Flexpert (Kouba et al., 2024). mdCATH (Mirarchi et al., 2024): 500 CATH domains at 320 K, the temperature closest to the ATLAS condition, scored under the scale fitted on ATLAS so the whole block tests transfer. *Slope* is the within-protein regression of prediction on measured RMSF, per-protein median, where 1.000 is perfect and lower means the model compresses the dynamic range. *MAE* is mean absolute error in ångströms and > 3 *Å* restricts it to residues whose *measured* RMSF exceeds 3 Å, the mobile regions a flexibility specification most often targets. Error columns use a fixed set of proteins that contain a mobile region (at least three residues above 5 Å; ATLAS 70, mdCATH 276), so every error column is the same proteins; correlation and slope use the full benchmark. *One independent MD replica* is a reproducibility reference, not an upper bound: each replica is scored against the mean of the remaining ones (three per protein on ATLAS, five on mdCATH) and averaged over replicas and proteins. *Structure only. †Structure + one-hot sequence, retrained on global RMSF. **Bold**: best. Underline: second.

Method	ATLAS (in distribution, 139 proteins)					mdCATH 320 K (OOD, 500 domains)				
	$r\uparrow$	$\rho\uparrow$	slope $\to 1$	MAE$\downarrow$	> 3 Å$\downarrow$	$r\uparrow$	$\rho\uparrow$	slope $\to 1$	MAE$\downarrow$	> 3 Å$\downarrow$
One independent MD replica	*0.895*	*0.898*	*0.967*	*0.709*	*1.552*	*0.893*	*0.890*	*0.975*	*0.473*	*1.355*
ProDy ANM (Eyal et al., 2006)	0.761	0.791	0.371	2.346	4.381	0.711	0.749	0.218	1.348	3.476
Flexpert-Seq (Kouba et al., 2024)	0.776	0.708	0.709	1.165	2.435	0.740	0.708	0.473	1.298	2.831
Flexpert-3D (Kouba et al., 2024)	0.824	0.762	0.736	1.168	2.308	0.793	0.731	0.500	1.270	2.755
BackFlip* (Viliuga et al., 2025)	0.862	0.829	0.673	0.930	2.029	0.825	0.804	0.427	1.319	2.963
BackFlip† (Viliuga et al., 2025)	0.841	0.809	0.763	0.901	1.869	0.810	0.783	0.492	1.197	2.657
HierEGT (ours, Stage 1)	0.852	0.836	0.830	0.868	1.704	0.817	0.814	0.524	**1.063**	2.494
FlexFormer (ours, Stage 1 + 2)	**0.870**	**0.843**	**0.872**	**0.846**	**1.642**	**0.845**	**0.819**	**0.558**	1.132	**2.477**

Best on both benchmarks, and calibrated where it counts. FlexFormer is best on every ATLAS metric and on four of five on mdCATH, the exception going to our own Stage 1 (Table 1). On correlation, the quantity this field reports, the leading methods are close on both benchmarks; on the two measurements a design loop actually consumes, they separate clearly. *Slope*, the within-protein regression of predicted on measured RMSF, is the fraction of a requested change a model reports back. Correlation is scale-invariant, so a model may rank every residue correctly yet compress a 5 Å loop to 3 Å. This is disqualifying for design: a specification is a magnitude, and a guidance loop halts once the predictor reports the target met, so a model returning half of each change stops at half the target while reporting success. FlexFormer returns 0.872 on ATLAS against 0.763 for the best baseline, and 0.558 against 0.500 OOD. All models compress OOD, in an order the correlation columns do not reveal: BackFlip is second-best on mdCATH correlation and second-worst on its calibration. *MAE above 3 Å* scores only the residues that genuinely move. Functional motion concentrates in mobile regions: loops gating cryptic pockets (Vajda et al., 2018; Meller et al., 2023), hinges coupling distant sites (Henzler-Wildman & Kern, 2007), and catalytic loops whose dynamics govern turnover (Corbella et al., 2023), so this is where a flexibility specification is typically directed. FlexFormer leads on both, 1.642 Å and 2.477 Å; on ATLAS its 0.227 Å margin over the strongest baseline is four times its 0.055 Å margin on overall MAE.

Table 2: **Response to a point substitution at fixed backbone.** *Fidelity* correlates the twenty single-substitution predictions at a position with the measured amino-acid flexibility scale π, averaged over all 139 ATLAS test chains and 2,771 positions and all 500 mdCATH domains and 10,000 positions at 320 K. π is estimated on the ATLAS training split alone, so it shares no protein with either benchmark. BackFlip† is structure with one-hot sequence (Viliuga et al., 2025); coordinate-only methods are omitted, undefined here.

Method	ATLAS fid.↑	mdCATH fid.↑
BackFlip†	0.057	0.059
Flexpert-3D	0.400	0.387
FlexFormer (ours)	**0.572**	**0.553**

Designers edit sequence, not coordinates. Protein engineering tunes flexibility by substitution at fixed fold: rigidifying mobile positions is a standard route to thermostability (Matthews, 1995; Hoppe & Schomburg, 2005; Del Amparo et al., 2023), while an over-stiffened catalytic loop loses turnover (Corbella et al., 2023). Methods predicting the perturbation effect of a single substitution, on stability or on binding (Pancotti et al., 2022; Deng et al., 2025; Jiao et al., 2025), score the mutant on the wild-type structure and treat the geometry as unchanged. A substitution does in fact repack its neighborhood, and the idealization is most consequential for flexibility, where the perturbed ensemble is itself the predicted quantity. It is nonetheless forced: no dataset pairs a point mutation with molecular dynamics of the mutant, and ATLAS contains only five usable near-identical pairs among 1,390 chains, so neither a mutated backbone nor a per-mutation target is available. We therefore hold the structural input at the wild-type backbone, vary only the sequence, and score against an aggregate, applying the identical protocol to every baseline. Under it a predictor reading coordinates alone returns one identical profile for all twenty candidates, a property of the function class that no benchmark holding sequence fixed can expose. The aggregate is $\pi \in \mathbb{R}^{20}$, how flexible each amino acid generally is, measured on ATLAS training data with no model involved: standardizing each protein within itself, $z_i = (y_i - \bar{y})/\mathrm{sd}(y)$, we set $\pi_a = \mathbb{E}_{\mathrm{prot}}[\mathbb{E}_{i:s_i=a} z_i]$, making π a statement about residue identity rather than about the proteins a residue occupies (it agrees at $r \geq 0.92$ across ATLAS splits and against mdCATH). With $v_a = f_\theta(S^{p\to a}, \mathcal{X})_p$ the prediction at a position edited to a, *fidelity* is $\phi_p = \mathrm{corr}(v - \bar{v}, \pi)$, one Pearson coefficient over twenty points, averaged over sampled positions. Measured this way, predicting the perturbation effect on flexibility appears to require two separate ingredients. One is a sequence representation carrying evolutionary constraint, which a one-hot encoding lacks entirely. The other is a readout that draws on it: Flexpert-3D shares our pretrained encoder, differing from FlexFormer only in the readout trained above it, yet leans on its structural channel rather than on sequence. A language model constrained by evolution already encodes which substitutions a fold tolerates; supervising the readout on measured fluctuation redirects that generic plausibility into a directed estimate of how a substitution changes flexibility.

4 Flexibility-Conditioned Backbone Design

FlexFormer's differentiability with respect to backbone Cα coordinates, $\nabla_{\mathcal{X}} f_\theta(S, \mathcal{X})$, is the property that makes flexibility an actionable signal during generation. The question is how to inject this signal without compromising the structural prior learned from PDB. FliPS (Viliuga et al., 2025) addresses this via conditional fine-tuning, backpropagating flexibility gradients through the generative model. This has three drawbacks: fine-tuning is expensive; the auxiliary flexibility loss corrupts the structural prior, producing designs that ESMFold cannot confidently refold; and the fixed conditional model requires screening large candidate pools at inference time. We instead leave the generative model entirely frozen and steer it at inference time via Stage-1 gradients, preserving the structural prior while enabling continuous flexibility conditioning through a single scalar λ.

MultiFlow (Campbell et al., 2024), a flow-matching co-design model that couples continuous SE(3) flows over backbone structure with a discrete sequence flow via Continuous Time Markov Chains (CTMCs) in a single joint model, ensures that structure and sequence evolve together during generation. This joint formulation means that guidance acts on a backbone distribution that is already aware of sequence context, providing a richer structural prior for flexibility-conditioned generation than structure-alone models. We adopt MultiFlow as our pretrained generative backbone. Sampling integrates an ODE from $t_{\min}$ to 1, where at each step the network g_ϕ predicts clean data $(\hat{\mathbf{T}}_0, \hat{R}_0, \hat{a}_0)$ from the noisy state, and an Euler step advances translations, rotations via geodesic interpolation on SO(3), and amino-acid types via a purity-weighted categorical step Yim et al. (2023); Lipman et al. (2022), with self-conditioning applied throughout. Given a desired per-residue flexibility profile $y^* \in \mathbb{R}^N$, we seek to generate backbone structures $\mathcal{X} \in \mathbb{R}^{N\times 3}$ whose predicted flexibility, as measured by the Stage-1 HierEGT encoder $f_\theta^{(1)}$, closely matches y^*. Formally, we sample from the conditional distribution

$$p(\mathcal{X} \mid y^*) \;\propto\; p(\mathcal{X}) \;\exp\!\big(-\lambda\, \mathcal{L}_{\text{guide}}(f_\theta^{(1)}(\mathcal{X}),\, y^*)\big),$$

where λ controls the strength of conditioning, without retraining or fine-tuning of g_ϕ.

4.1 Inference-Time Guidance via Differentiable RMSF Prediction

We condition the MultiFlow ODE trajectory on the target flexibility profile y^* without retraining or fine-tuning of the flow model. The key insight is that the Stage-1 HierEGT encoder $f_\theta^{(1)}$ is differentiable with respect to backbone C_α coordinates, providing a gradient signal that steers the trajectory toward the target profile.

Guidance step. At each ODE step with $t \geq t_{\text{start}}$, the flow network g_ϕ produces a clean translation prediction $\hat{\mathbf{T}}_0 \in \mathbb{R}^{N\times 3}$, which we treat as a proxy for the final backbone. We compute $\hat{y} = f_\theta^{(1)}(\hat{\mathbf{T}}_0)$ and evaluate the guidance loss as the negative Pearson correlation,

$$\mathcal{L}_{\text{guide}}(\hat{y},\, y^*) = -\frac{(\hat{y} - \bar{\hat{y}})^\top (y^* - \bar{y}^*)}{\|\hat{y} - \bar{\hat{y}}\|\, \|y^* - \bar{y}^*\| + \varepsilon}, \tag{1}$$

chosen for its scale- and shift-invariance. The gradient $\mathbf{g} = \nabla_{\hat{\mathbf{T}}_0} \mathcal{L}_{\text{guide}}$ is clipped per residue as $\tilde{g}_i = \mathbf{g}_i \cdot \min(1,\, \delta/(\|\mathbf{g}_i\| + \varepsilon))$ to prevent any single residue from destabilising the trajectory, and the guided prediction is $\hat{\mathbf{T}}_0^{\text{guided}} = \hat{\mathbf{T}}_0 - \lambda\, \tilde{\mathbf{g}}$, where $\lambda > 0$ is the guidance scale. This replaces $\hat{\mathbf{T}}_0$ in the Euler update $\mathbf{T}_{t+\Delta t} = \mathbf{T}_t + \Delta t \cdot (\hat{\mathbf{T}}_0^{\text{guided}} - \mathbf{T}_t)/(1-t)$ and is also fed back as the self-conditioning input at the next step. Rotation and amino-acid type updates proceed exactly as in the unguided ODE; guidance acts exclusively through the translational component. Guidance is applied only for $t \geq t_{\text{start}}$, leaving the early noisy portion of the trajectory unperturbed, where clean predictions are unreliable Dhariwal & Nichol (2021). The gradient is computed by a single backward pass through the frozen HierEGT encoder only, adding approximately one forward-backward pass of overhead per ODE step; full hyperparameter details and the complete sampling algorithm are provided in Appendix B.

4.2 Evaluation

Setup. We evaluate on two molecular dynamics benchmarks: ATLAS Vander Meersche et al. (2024), using the 139-protein topology-split test set, and mdCATH Mirarchi et al.

(2024), evaluated at 320 K on 500 domains, of which 464 meet the length constraint below. We compare against FliPS Viliuga et al. (2025), the current state-of-the-art flexibility-conditioned design method, and Unguided MultiFlow ($\lambda = 0$). Since FliPS only operates on proteins of length $L \in [60, 512]$, all comparisons are restricted to the $n = 126$ proteins satisfying this constraint; our method imposes no such restriction. We report four metrics: S2 Pearson (Pearson correlation between an independent T5-based RMSF evaluator Kouba et al. (2024) and target profile y^*), SC RMSD (self-consistency RMSD: ProteinMPNN Dauparas et al. (2022) redesign → ESMFold Lin et al. (2023) → Kabsch RMSD to generated backbone), and pLDDT (mean ESMFold confidence on the redesigned sequence), and runtime (seconds). For FliPS, S2 is evaluated on the ProteinMPNN-redesigned sequence (backbone → ProteinMPNN → best by pLDDT → T5).

Table 3: Flexibility-conditioned backbone generation on ATLAS ($n = 126$, $L \in [60, 512]$) and mdCATH ($n = 464$, $L \in [60, 512]$, all 5 splits). S2 Pearson is evaluated on ProteinMPNN-redesigned sequences for all methods except †, which uses the MultiFlow co-designed sequence. Runtime in seconds per protein on a single GPU; FliPS runtime reflects 50-candidate generation with BackFlip screening. **Bold**: best among flexibility-conditioned methods. FlexFlow is 26× faster than FliPS on ATLAS and 15× on mdCATH.

	ATLAS ($n = 126$)				mdCATH ($n = 464$)			
Method	S2 $r \uparrow$	scRMSD (Å)↓	pLDDT ↑	Runtime (s)↓	S2 $r \uparrow$	scRMSD (Å)↓	pLDDT ↑	Runtime (s)↓
Unguided MultiFlow†	0.402	9.94	83.7	11.0	0.420	6.64	87.7	10.0
Unguided MultiFlow	0.418	9.94	83.7	11.0	0.432	6.64	87.7	10.0
RMSF-conditional design (ProteinMPNN sequences)								
FliPS Viliuga et al. (2025)	**0.799**	12.32	64.2	416.9	0.732	**7.58**	**84.4**	191.8
FlexFlow (ours)	0.753	**9.35**	**75.2**	**15.8** (26x)	**0.757**	8.06	78.1	**12.4** (15x)

Results. Table 3 reports results at our selected operating point $\lambda = 400$. We evaluate on the same 139-protein ATLAS test set and 500 mdCATH domains (five splits of 100 domains each) used in our FlexFormer evaluation (Section 3.1), restricting both benchmarks to proteins with $L \in [60, 512]$ to match the length range on which FliPS was trained, yielding $n = 126$ ATLAS proteins and $n = 464$ mdCATH domains for the comparison. All S2 Pearson values are evaluated using ProteinMPNN-redesigned sequences for both methods, ensuring a fair comparison of the sequence–flexibility relationship independently of the generation procedure.

On the **in-distribution** ATLAS benchmark, FliPS achieves the highest S2 Pearson correlation (0.799 vs. 0.753), reflecting its dedicated conditional training on ATLAS-derived flexibility profiles. However, this comes at severe cost to structural quality: FliPS produces a mean pLDDT of 64.2, which is borderline unstructured by ESMFold's own confidence scale, and an SC RMSD of 12.32 Å, compared to 75.2 and 9.35 Å for our method. A generated protein that ESMFold cannot confidently fold is of limited practical utility regardless of its predicted flexibility correlation. This degradation is consistent with FliPS's training procedure, in which the flexibility auxiliary loss directly modifies the flow vector field via gradients backpropagated through the predicted clean structure $\hat{\mathbf{T}}_0$ Viliuga et al. (2025), corrupting the structural prior learned from PDB rather than leaving it intact at inference time. Our method maintains substantially better structural integrity while achieving competitive flexibility quality, and provides a continuous flexibility–designability trade-off controlled by the single scalar λ, a degree of control that FliPS, as a fixed conditional model, cannot offer.

On the **out-of-distribution** mdCATH benchmark, which features longer trajectories (464 ns vs. 100 ns), shorter proteins (137 vs. 236 residues on average), and a simulation temperature (320 K) different from the 300 K used during both HierEGT and FliPS training, our method outperforms FliPS on flexibility correlation (S2 Pearson 0.757 vs. 0.732, $p < 0.001$) across all 464 domains while maintaining competitive structural consistency (SC RMSD 8.06 vs. 7.58 Å). FliPS retains a pLDDT advantage on mdCATH (84.4 vs. 78.1). Both methods score higher here than on ATLAS, consistent with mdCATH's shorter domains (137 vs. 236

residues on average), which fold more confidently. What length does not explain is the spread: our pLDDT moves by three points across the two benchmarks (75.2 to 78.1) while FliPS moves by twenty (64.2 to 84.4), indicating that inference-time guidance delivers more uniform structural quality than dedicated conditional training. The advantage extends to runtime: because gradient-based guidance steers each backbone directly toward the target, a single sample suffices without the post-hoc screening that FliPS requires, yielding a 15–26× wall-clock speedup (Table 3; full length-stratified breakdown in Appendix D).

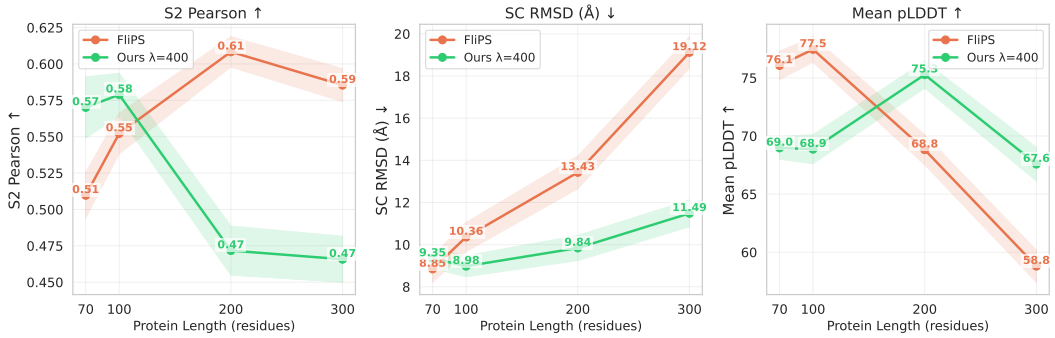


Figure 2: **De novo flexibility-conditioned design across protein lengths.** Comparison of FlexFlow ($\lambda = 400$) and FliPS on randomly sampled RMSF profiles following the RFdiffusion evaluation protocol, across protein lengths $L \in \{70, 100, 200, 300\}$ ($n = 100$ per length). Shaded regions denote standard error. FlexFlow achieves substantially lower SC RMSD at lengths 100, 200, 300 and similar SC RMSD at 70, indicating better structural designability, while maintaining competitive flexibility correlation (S2 Pearson). Higher pLDDT at $L \geq 200$ further confirms that our method generates more confidently foldable structures than FliPS as protein length increases.

De novo design with random flexibility profiles. To evaluate FlexFlow in a fully *de novo* setting, we follow the RFdiffusion evaluation protocol Watson et al. (2023) and generate proteins of varying lengths ($L \in \{70, 100, 200, 300\}$) conditioned on randomly sampled RMSF profiles unseen during training. For each length we generate $n = 100$ backbones and compare against FliPS, the current state-of-the-art flexibility-conditioned design method. All S2 Pearson values are evaluated using ProteinMPNN-redesigned sequences for both methods. Results are summarized in Figure 2. At shorter lengths ($L = 70, 100$), FlexFlow achieves higher S2 Pearson correlation than FliPS (0.57 vs. 0.51 at $L = 70$; 0.58 vs. 0.55 at $L = 100$), while maintaining comparable SC RMSD and pLDDT. At longer lengths ($L = 200, 300$), FliPS achieves higher S2 Pearson correlation (0.61 vs. 0.47 at $L = 200$; 0.59 vs. 0.47 at $L = 300$), consistent with its dedicated conditional training on flexibility profiles. However, this comes at a severe cost to structural quality: FliPS produces SC RMSD of 19.12 Å at $L = 300$ compared to 11.49 Å for our method, and mean pLDDT of 58.8 vs. 67.6, indicating that FliPS increasingly sacrifices structural validity for flexibility optimization as protein length grows. At $L \geq 100$, FlexFlow achieves substantially lower SC RMSD than FliPS (e.g. 9.84 Å vs. 13.43 Å at $L = 200$; 11.49 Å vs. 19.12 Å at $L = 300$), and higher pLDDT at $L \geq 200$ (75.3 vs. 68.8 at $L = 200$; 67.6 vs. 58.8 at $L = 300$), demonstrating that inference-time guidance preserves structural designability even as protein length increases. These results reinforce the conclusion that inference-time guidance on a pretrained flow matching model generalizes more robustly to the challenging de novo design setting than conditional training.

MD validation of generated backbones. To validate that FlexFlow-generated backbones realize the specified flexibility profiles under physical simulation, we performed explicit-solvent MD simulations (3 replicas × 100 ns each) on 40 generated backbones spanning three categories: 10 ATLAS test proteins, 10 mdCATH domains, and 20 hand-drawn flexibility profiles at lengths $L \in \{70, 100, 200, 300\}$. For each generated backbone, ProteinMPNN-redesigned sequences were simulated and per-residue RMSF was compared against the target profile. Figure 3 shows representative examples from each category alongside the distribution of Pearson correlations across all validated designs. ATLAS and mdCATH examples achieve strong agreement with target profiles (mean $r = 0.82$ and 0.90, respectively), confirming

that FlexFlow generates physically realistic flexibility. Hand-drawn profiles, which specify arbitrary user-defined RMSF patterns on de novo backbones with no natural reference, achieve a mean $r = 0.51$, reflecting the difficulty of realizing unconstrained specifications. Full per-protein profiles are provided in Appendix Figures 5, 6, and 7.

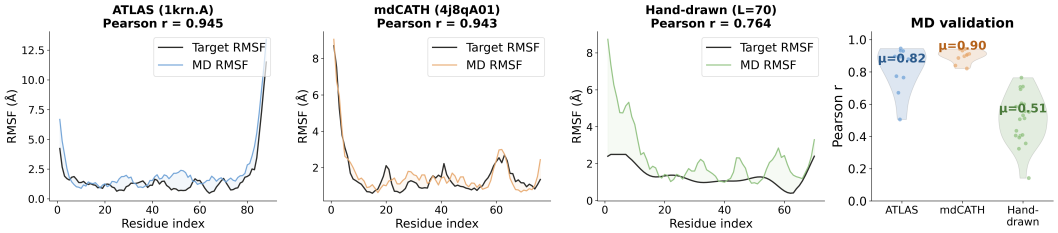


Figure 3: **MD validation of FlexFlow-generated backbones.** (Left three panels) Representative RMSF profiles for the best-performing example from each dataset: ATLAS, mdCATH, and hand-drawn (L=70). Black curves show the target RMSF; colored curves show MD-derived RMSF averaged over three replicas. (Right panel) Distribution of Pearson correlations between MD-derived and target RMSF across all 40 validated designs.

Qualitative examples. Figure 4 shows representative FlexFlow-generated backbones alongside reference structures for one ATLAS and one mdCATH example. In both cases, FlexFlow generates a structurally distinct backbone whose flexibility profile closely matches the target, as confirmed by MD simulation (Pearson $r = 0.896$ for ATLAS 1u07.A and $r = 0.942$ for mdCATH 1tc3C00). Both generated structures are highly designable and confidently foldable, demonstrating that FlexFlow produces novel, well-structured backbones that faithfully realize the specified flexibility profile.

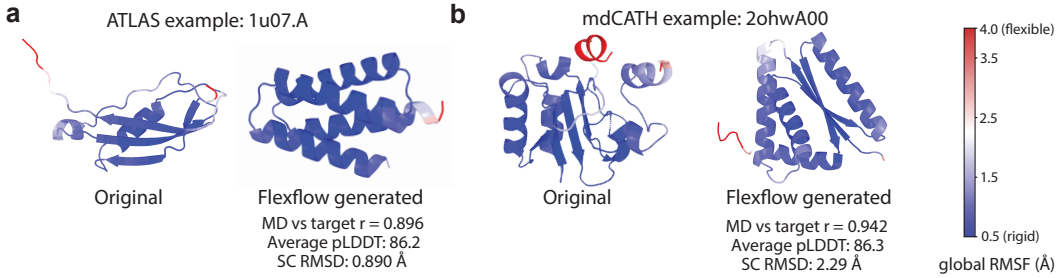


Figure 4: **Representative FlexFlow-generated backbones colored by MD RMSF.** **(a)** ATLAS example (1u07.A): original crystal structure (left) and FlexFlow-generated backbone (right), colored by per-residue RMSF from MD simulation ($r = 0.896$, SC RMSD $= 0.890$ Å, pLDDT $= 86.2$). **(b)** mdCATH example (1tc3C00): the original structure (left) and FlexFlow-generated backbone (right) ($r = 0.942$, SC RMSD $= 2.29$ Å, pLDDT $= 86.3$). Blue: rigid (low RMSF); red: flexible (high RMSF). FlexFlow generates structurally novel backbones with distinct topologies that faithfully reproduce the target flexibility profile while maintaining high designability.

5 Conclusion

We introduced FlexFlow, a flexibility-conditioned backbone design framework that uses inference-time guidance on a frozen flow-matching co-design model. FlexFlow builds on FlexFormer, a differentiable two-stage RMSF predictor we developed, which attains the best reported ATLAS accuracy and retains its ranking on mdCATH. Beyond profile accuracy, we introduced a measurement of how a predictor responds to single-point mutations, which every benchmark in current use is blind to because all of them hold sequence fixed, and on which FlexFormer leads every method that reads both sequence and structure. FlexFlow exceeds FliPS on flexibility correlation out of distribution while preserving structural designability, runs 15–26× faster, offers continuous control over the flexibility-designability trade-off, and is validated by MD simulation on 40 generated backbones.

AI use statement

We used generative AI tools solely to improve the grammar, clarity, and readability of the manuscript. We did not use them to develop research ideas or hypotheses, design experiments, implement methods, analyze data, interpret results, or formulate scientific claims. The authors reviewed and edited the AI-assisted text, verified that it accurately represents the research, and take full responsibility for the final manuscript.

Ethics statement

This work uses publicly available datasets and pretrained models for computational research. It does not involve human subjects, personally identifiable information, or private or sensitive data. We report the provenance of the datasets and models used and evaluate the proposed method within the scope described in the paper. The authors are responsible for the integrity of the experiments and for accurately representing the capabilities and limitations of the method.

Reproducibility statement

The main text describes the proposed method and experimental design. The appendix provides further details on the datasets and preprocessing, data splits, model and baseline configurations, training procedures, evaluation metrics, and implementation settings. We also report the experimental comparisons and analyses needed to assess the results. Together, these materials are intended to support independent reproduction of the findings.

References

Ivet Bahar, Ali Rana Atilgan, and Burak Erman. Direct evaluation of thermal fluctuations in proteins using a single-parameter harmonic potential. *Folding and Design*, 2(3):173–181, 1997.

Stephen J Benkovic, Gordon G Hammes, and Sharon Hammes-Schiffer. Free-energy landscape of enzyme catalysis. *Biochemistry*, 47(11):3317–3321, 2008.

Nadav Brandes, Grant Goldman, Charlotte H Wang, Chun Jimmie Ye, and Vasilis Ntranos. Genome-wide prediction of disease variant effects with a deep protein language model. *Nature genetics*, 55(9):1512–1522, 2023.

Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. *arXiv preprint arXiv:2402.04997*, 2024.

Marina Corbella, Gaspar P Pinto, and Shina CL Kamerlin. Loop dynamics and the evolution of enzyme activity. *Nature Reviews Chemistry*, 7(8):536–547, 2023.

Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022.

Roberto Del Amparo, Luis Daniel González-Vázquez, Laura Rodríguez-Moure, Ugo Bastolla, and Miguel Arenas. Consequences of genetic recombination on protein folding stability. *Journal of Molecular Evolution*, 91(1):33–45, 2023.

Arthur Deng, Karsten Householder, Fang Wu, Sebastian Thrun, K Christopher Garcia, and Brian Trippe. Predicting mutational effects on protein binding from folding energy. *arXiv preprint arXiv:2507.05502*, 2025.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, et al. Prottrans: Toward understanding the language of life through self-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7112–7127, 2021.

Eran Eyal, Lee-Wei Yang, and Ivet Bahar. Anisotropic network model: systematic evaluation and a new web interface. *Bioinformatics*, 22(21):2619–2627, 2006.

Cameron J Glasscock, Robert J Pecoraro, Ryan McHugh, Lindsey A Doyle, Wei Chen, Olivier Boivin, Beau Lonnquist, Emily Na, Yuliya Politanska, Hugh K Haddox, et al. Computational design of sequence-specific dna-binding proteins. *Nature Structural & Molecular Biology*, 32(11):2252–2261, 2025.

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

Brian L Hie, Varun R Shanker, Duo Xu, Theodora UJ Bruun, Payton A Weidenbacher, Shaogeng Tang, Wesley Wu, John E Pak, and Peter S Kim. Efficient evolution of human antibodies from general protein language models. *Nature biotechnology*, 42(2):275–283, 2024.

Scott A Hollingsworth and Ron O Dror. Molecular dynamics simulation for all. *Neuron*, 99 (6):1129–1143, 2018.

Scott A Hollingsworth, Brendan Kelly, Celine Valant, Jordan Arthur Michaelis, Olivia Mastromihalis, Geoff Thompson, AJ Venkatakrishnan, Samuel Hertig, Peter J Scammells, Patrick M Sexton, et al. Cryptic pocket formation underlies allosteric modulator selectivity at muscarinic gpcrs. *Nature Communications*, 10(1):3289, 2019.

Christian Hoppe and Dietmar Schomburg. Prediction of protein thermostability with a direction-and distance-dependent knowledge-based potential. *Protein Science*, 14(10): 2682–2692, 2005.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2022.

Jing Huang, Sarah Rauscher, Grzegorz Nawrocki, Ting Ran, Michael Feig, Bert L De Groot, Helmut Grubmüller, and Alexander D MacKerell Jr. Charmm36m: an improved force field for folded and intrinsically disordered proteins. *Nature methods*, 14(1):71–73, 2017.

Xiaoran Jiao, Weian Mao, Wengong Jin, Peiyuan Yang, Hao Chen, and Chunhua Shen. Boltzmann-aligned inverse folding model as a predictor of mutational effects on protein-protein interactions. In *International Conference on Learning Representations*, volume 2025, pp. 18209–18226, 2025.

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.

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.

Ziang Li and Yunan Luo. Generalizable and scalable protein stability prediction with rewired protein generative models. *Nature Communications*, 2025.

Qinghua Liao, Yashraj Kulkarni, Ushnish Sengupta, Dušan Petrović, Adrian J Mulholland, Marc W van der Kamp, Birgit Strodel, and Shina Caroline Lynn Kamerlin. Loop motion in triosephosphate isomerase is not a simple open and shut case. *Journal of the American Chemical Society*, 140(46):15889–15903, 2018.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Sal Candido, et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. *BioRxiv*, 2022:500902, 2022.

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

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Yunan Luo, Guangde Jiang, Tianhao Yu, Yang Liu, Lam Vo, Hantian Ding, Yanan Su, Wesley Wei Qian, Huimin Zhao, and Jian Peng. Ecnet is an evolutionary context-integrated deep learning framework for protein engineering. *Nature communications*, 12(1):5743, 2021.

Brian W Matthews. Studies on protein stability with t4 lysozyme. *Advances in protein chemistry*, 46:249–278, 1995.

Joshua Meier, Roshan Rao, Robert Verkuil, Jason Liu, Tom Sercu, and Alexander Rives. Language models enable zero-shot prediction of the effects of mutations on protein function. *Advances in neural information processing systems*, 34:29287–29303, 2021.

Artur Meller, Michael Ward, Jonathan Borowsky, Meghana Kshirsagar, Jeffrey M Lotthammer, Felipe Oviedo, Juan Lavista Ferres, and Gregory R Bowman. Predicting locations of cryptic pockets from single protein structures using the pocketminer graph neural network. *Nature communications*, 14(1):1177, 2023.

Grover Paul Miller and Stephen J Benkovic. Stretching exercises—flexibility in dihydrofolate reductase catalysis. *Chemistry & biology*, 5(5):R105–R113, 1998.

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.

Martin Pacesa, Lennart Nickel, Christian Schellhaas, Joseph Schmidt, Ekaterina Pyatova, Lucas Kissling, Patrick Barendse, Jagrity Choudhury, Srajan Kapoor, Ana Alcaraz-Serna, et al. Bindcraft: one-shot design of functional protein binders. *BioRxiv*, pp. 2024–09, 2024.

Corrado Pancotti, Silvia Benevenuta, Giovanni Birolo, Virginia Alberini, Valeria Repetto, Tiziana Sanavia, Emidio Capriotti, and Piero Fariselli. Predicting protein stability changes upon single-point mutation: a thorough comparison of the available tools on a new dataset. *Briefings in Bioinformatics*, 23(2):bbab555, 2022.

Arvind Pillai, Abbas Idris, Annika Philomin, Connor Weidle, Rebecca Skotheim, Philip JY Leung, Adam Broerman, Cullen Demakis, Andrew J Borst, Florian Praetorius, et al. De novo design of allosterically switchable protein assemblies. *Nature*, 632(8026):911–920, 2024.

Roberto Scandurra, Valerio Consalvi, Roberta Chiaraluce, Laura Politi, and Paul C Engel. Protein thermostability in extremophiles. *Biochimie*, 80(11):933–941, 1998.

David E Shaw, Paul Maragakis, Kresten Lindorff-Larsen, Stefano Piana, Ron O Dror, Michael P Eastwood, Joseph A Bank, John M Jumper, John K Salmon, Yibing Shan, et al. Atomic-level characterization of the structural dynamics of proteins. *Science*, 330 (6002):341–346, 2010.

Lars Skjaerven, Siv M Hollup, and Nathalie Reuter. Normal mode analysis for proteins. *Journal of Molecular Structure: THEOCHEM*, 898(1-3):42–48, 2009.

Samuel Sledzieski, Meghana Kshirsagar, Minkyung Baek, et al. Democratizing protein language models with parameter-efficient fine-tuning. *PNAS*, 2024.

Zhoutong Sun, Qian Liu, Ge Qu, Yan Feng, and Manfred T Reetz. Utility of b-factors in protein science: interpreting rigidity, flexibility, and internal motion and engineering thermostability. *Chemical reviews*, 119(3):1626–1665, 2019.

Sandor Vajda, Dmitri Beglov, Amanda E Wakefield, Megan Egbert, and Adrian Whitty. Cryptic binding sites on proteins: definition, detection, and druggability. *Current opinion in chemical biology*, 44:1–8, 2018.

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.

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.

Simon Wagner, Leif Seute, Vsevolod Viliuga, Nicolas Wolf, Frauke Gräter, and Jan Stühmer. Generating highly designable proteins with geometric algebra flow matching. *Advances in Neural Information Processing Systems*, 37:77987–78026, 2024.

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, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, 2023.

Jason Yim, Brian L Trippe, Valentin De Bortoli, Emile Mathieu, Arnaud Doucet, Regina Barzilay, and Tommi Jaakkola. Se (3) diffusion model with application to protein backbone generation. *arXiv preprint arXiv:2302.02277*, 2023.

Edward W Yu and Daniel E Koshland Jr. Propagating conformational changes over long (and short) distances in proteins. *Proceedings of the National Academy of Sciences*, 98 (17):9517–9520, 2001.

Vinicius Zambaldi, David La, Alexander E Chu, Harshnira Patani, Amy E Danson, Tristan OC Kwan, Thomas Frerix, Rosalia G Schneider, David Saxton, Ashok Thillaisundaram, et al. De novo design of high-affinity protein binders with alphaproteo. *arXiv preprint arXiv:2409.08022*, 2024.

Junming Zhao, Chao Zhang, and Yunan Luo. Contrastive fitness learning: Reprogramming protein language models for low-n learning of protein fitness landscape. In *International Conference on Research in Computational Molecular Biology*, pp. 470–474. Springer, 2024.

Kirill Zinovjev, Paul Guénon, Carlos A Ramos-Guzmán, J Javier Ruiz-Pernía, Damien Laage, and Iñaki Tuñón. Activation and friction in enzymatic loop opening and closing dynamics. *Nature Communications*, 15(1):2490, 2024.

Appendix

Contents

A FlexFormer: Architecture and Training Details **16**

A.1 FlexFormer Architecture 16

A.2 Training and implementation details 16

B FlexFlow: Implementation and Evaluation Details **18**

C MD Validation: Full RMSF Profiles **19**

D Runtime Analysis **24**

E Reproducibility Statement **24**

A FlexFormer: Architecture and Training Details

A.1 FlexFormer Architecture

HierEGT consists of 4 local k-NN graph neural network (GNN) blocks interleaved with 2 global read-only Edge-augmented Graph Transformer (EGT) blocks, following a 2-local $\rightarrow$ 1-global $\rightarrow$ 2-local $\rightarrow$ 1-global schedule. The model has 4.73M parameters.

Input encoding. Given C_α coordinates, we first construct per-residue local frames (t_i, R_i) using the backbone geometry, then compute SE(3)-equivariant edge features via radial basis function (RBF) distance encoding and relative orientation features. Node features are initialized as the mean of incoming edge features concatenated with a normalised sequence position encoding. Edge features are fixed after initialization and never updated during the forward pass.

Local k-NN GNN blocks. Each local block computes multi-head attention restricted to the $k = 12$ nearest neighbours in Euclidean distance. Edge features enter only as additive attention biases and are not updated. The sparse neighbourhood prevents any residue from attending to distant partners at this stage, ensuring that learned representations reflect local geometric context: backbone torsion angles, secondary structure hydrogen bonding geometry, and short-range packing. Because the receptive field is fixed at 12 residues regardless of protein length or simulation condition, the representations learned by these blocks are insensitive to the global topology of the protein and transfer readily across protein families. Each local block uses 4 attention heads with per-head dimension $d_h = 32$, followed by a sigmoid-gated feed-forward network with hidden dimension $4d_{\text{node}} = 1024$ and GELU activations.

Global read-only EGT blocks. Each global block computes full $L \times L$ multi-head attention, using the fixed edge features as additive biases. Critically, the edge representation is never updated: the global block reads geometric context from the pair encoding but does not write back to it. This is the key design choice distinguishing HierEGT from architectures such as CFA Wagner et al. (2024) or IPA Jumper et al. (2021), in which the pair representation is iteratively refined across layers. In those architectures, the pair matrix accumulates dataset-specific co-variation statistics across layers, encoding correlations between residue pairs that are specific to the training distribution. By contrast, HierEGT's pair encoding remains a fixed geometric description of the input throughout the forward pass, and the global blocks use it only to bias attention toward geometrically relevant residue pairs.

Interleaving rationale. The 2-local $\rightarrow$ 1-global schedule is deliberate. Local blocks first build rich per-residue representations grounded in neighbourhood geometry. The global block then aggregates these representations across the full sequence, using the geometric pair encoding to weight contributions by spatial proximity. A second round of local refinement then re-contextualises each residue's representation in light of the global signal, before a final global aggregation. This hierarchical processing mirrors the physical organisation of protein flexibility: local secondary structure geometry determines the baseline rigidity of each residue, while long-range contacts and domain topology modulate it.

A.2 Training and implementation details

Edge feature construction. SE(3)-equivariant edge features $e_{ij}^{(0)} \in \mathbb{R}^{47}$ are computed from inter-residue geometry: 32-dimensional RBF distance features over the range 0–20 Å, the unit displacement vector $\hat{r}_{ij}$ expressed in each residue's local C_α frame (2×3 dimensions), and the relative rotation matrix between frames (9 dimensions). Node features are initialized by averaging incoming edge features over all neighbours, augmented with a normalised sequence-position scalar, and projected to $d_{\text{node}} = 256$ by a two-layer MLP with GELU activations. Edge features are projected to $d_{\text{edge}} = 64$ and remain fixed for the duration of the forward pass.

Readout head. After all message-passing blocks, a three-layer MLP ($256 \to 128 \to 64 \to 1$) with LayerNorm input normalisation and Softplus output activation maps each node embedding $h_i^{(L)} \in \mathbb{R}^{256}$ to a non-negative scalar $\hat{y}_i^{(1)}$

Training. Stage 1 is optimised on the ATLAS training split with the combined loss

$$\mathcal{L}_1 = \lambda_{\text{MSE}} \mathcal{L}_{\text{MSE}} + \lambda_r (1 - r(\hat{y}, y)), \tag{2}$$

where $r(\cdot, \cdot)$ denotes Pearson correlation, $\lambda_{\text{MSE}} = 0.5$, and $\lambda_r = 1.0$. We use AdamW (lr $= 3 \times 10^{-4}$, weight decay 10^{-4}) with gradient norm clipping at 1.0, a batch size of 4 proteins, and a cosine annealing learning rate schedule over 50 epochs. Early stopping is applied on validation Pearson correlation with a patience of 15 evaluation steps, evaluated every 200 gradient steps. Training converges within approximately 40k gradient steps on a single NVIDIA A100 GPU.

Stage 2 adaptor. Stage 2 initializes from the pretrained FlexPert-3D weights Kouba et al. (2024), freezes the entire ProtT5-XL-U50 encoder, and fine-tunes only a 1D CNN adaptor head (two convolutional layers of kernel size 3 with GELU activation and hidden dimension 128, projecting to the T5 embedding dimension of 1024) totalling 165K trainable parameters. The Stage 1 RMSF estimates are injected as a per-token scalar signal added to the corresponding T5 token embeddings before encoding. The same loss and optimiser configuration as Stage 1 are used, and training converges within approximately 15k gradient steps on a single NVIDIA A100 GPU.

B FlexFlow: Implementation and Evaluation Details

Algorithm 1 summarises the flexibility-guided ODE sampling procedure. The key modification to the standard MultiFlow sampling loop is the insertion of a single guidance correction (lines 4–6) that steers the translational component toward the target flexibility profile while leaving rotations and amino-acid types to evolve under the unmodified flow. Because MultiFlow jointly co-designs structure and sequence, the guided translations implicitly constrain the sequence trajectory through self-conditioning, achieving structure–sequence consistency without explicit sequence-level guidance.

Algorithm 1 FlexFlow: Flexibility-Guided ODE Sampling

Require: Length N, target profile $\mathbf{y}^* \in \mathbb{R}^N$, guidance scale λ, start time t_{start}, clip threshold δ, steps K
Frozen HierEGT $f_\theta^{(1)}$, frozen MultiFlow g_ϕ

1: Sample $(\mathbf{T}_0, \mathbf{R}_0, \mathbf{a}_0) \sim p_0$ ▷ Gaussian × Uniform SO(3) × Uniform $\mathcal{A}$
2: **for** $k = 0, \ldots, K-1$ **do** with time t_k, step size Δt
3: $(\hat{\mathbf{T}}_0, \hat{\mathbf{R}}_0, \hat{\mathbf{a}}_0) \leftarrow g_\phi(\mathbf{T}_t, \mathbf{R}_t, \mathbf{a}_t, t_k)$ ▷ Clean prediction, no grad
4: **if** $\lambda > 0$ and $t_k \geq t_{\text{start}}$ **then**
5: $\tilde{\mathbf{g}} \leftarrow \textsc{ClipPerResidue}\big(\nabla_{\hat{\mathbf{T}}_0}\big[-\text{Pearson}(f_\theta^{(1)}(\hat{\mathbf{T}}_0), \mathbf{y}^*)\big],\ \delta\big)$
6: $\hat{\mathbf{T}}_0 \leftarrow \hat{\mathbf{T}}_0 - \lambda\, \tilde{\mathbf{g}}$ ▷ Guided translation
7: **end if**
8: $\mathbf{T}_{t+\Delta t} \leftarrow \mathbf{T}_t + \Delta t \cdot (\hat{\mathbf{T}}_0 - \mathbf{T}_t) \,/\, (1 - t_k)$ ▷ Translation Euler step
9: $\mathbf{R}_{t+\Delta t} \leftarrow \text{SO3}\textsc{EulerStep}(\mathbf{R}_t, \hat{\mathbf{R}}_0, \Delta t, t_k)$ ▷ Rotation step (unguided)
10: $\mathbf{a}_{t+\Delta t} \leftarrow \textsc{CatEulerStep}(\mathbf{a}_t, \hat{\mathbf{a}}_0, \Delta t, t_k)$ ▷ Sequence step (unguided)
11: Update self-conditioning: $\mathbf{T}^{\text{sc}} \leftarrow \hat{\mathbf{T}}_0, \quad \mathbf{a}^{\text{sc}} \leftarrow \hat{\mathbf{a}}_0$
12: **end for**
13: **return** backbone from final $(\hat{\mathbf{T}}_0, \hat{\mathbf{R}}_0)$, sequence from $\hat{\mathbf{a}}_0$

The guidance gradient (line 4) is computed by a single backward pass through the frozen HierEGT encoder; the MultiFlow forward pass runs under `no_grad`. Per-residue clipping $\tilde{\mathbf{g}}_i = \mathbf{g}_i \cdot \min(1, \delta / \|\mathbf{g}_i\|)$ prevents any single residue from dominating the correction. Note that guidance acts *exclusively* on the translational component $\hat{\mathbf{T}}_0$; rotations and amino-acid logits follow the unmodified MultiFlow ODE. Nonetheless, the self-conditioning mechanism (line 11) propagates the guided structure back into subsequent network evaluations, so the co-designed sequence implicitly adapts to the guided backbone geometry at every step.

Hyperparameters. Guidance uses $N = 200$ ODE steps activated from $t_{\text{start}} = 0.3$ onward, per-residue gradient-norm clipping threshold $\delta = 0.5$, and numerical stability constant $\varepsilon = 10^{-8}$. The Stage-2 ProtT5 adaptor is excluded from guidance as the amino-acid sequence evolves stochastically alongside structure during sampling, making differentiation through the sequence encoder ill-defined. We investigate $\lambda \in \{20, 50, 100, 400\}$ and use $\lambda = 400$ for all benchmark comparisons, balancing flexibility fidelity against structural designability. All experiments use a single NVIDIA A100 GPU.

Post-generation evaluation. Each generated backbone undergoes sequence redesign with ProteinMPNN (8 sequences, $T = 0.1$). For each redesigned sequence, we (i) predict RMSF using the independent Stage-2 FlexFormer evaluator and compute S2 Pearson correlation against the target $\mathbf{y}^*$, selecting the sequence with highest S2, and (ii) refold with ESMFold to obtain SC RMSD (Kabsch RMSD between generated and refolded backbone atoms N/Cα/C) and mean pLDDT. A structure is deemed *designable* if SC RMSD ≤ 2.0 Å.

C MD Validation: Full RMSF Profiles

This appendix details the molecular dynamics (MD) protocol used to physically validate FlexFlow-generated backbones (Figure 3) and provides the complete per-protein RMSF profiles for all 40 validated designs. For every generated backbone we simulated the ProteinMPNN-redesigned sequence in explicit solvent and compared the resulting per-residue RMSF against the target profile originally passed to FlexFlow as the conditioning signal. All simulations were carried out on NCSA DeltaAI, an NVIDIA GH200 Grace Hopper system (ARM Grace CPUs, H100 GPUs, CUDA architecture 90), using GROMACS 2026.1 compiled from source with CUDA GPU offloading against the default DeltaAI toolchain (gcc 14.2.0, CUDA 12.9, CMake 3.30.2).

Rationale for explicit-solvent MD. The Stage 1 HierEGT predictor and the downstream S2 evaluator are both learned proxies for ground-truth flexibility; agreement between them does not by itself establish that FlexFlow designs would behave as intended under physical simulation. To close this gap we perform the same explicit-solvent MD protocol used to generate the ATLAS reference data Vander Meersche et al. (2024), treating the resulting per-residue RMSF as the most reliable available measurement of the design's actual flexibility. This provides an independent, physics-based check on the full pipeline (FlexFlow generation → ProteinMPNN redesign → MD simulation) rather than on any individual learned component.

System selection. We validated 40 generated backbones spanning three categories: (i) 10 designs conditioned on RMSF profiles from held-out ATLAS test proteins, (ii) 10 designs conditioned on profiles from held-out mdCATH domains, and (iii) 20 designs conditioned on hand-drawn synthetic RMSF profiles at lengths $L \in \{70, 100, 200, 300\}$ (5 per length). The ATLAS and mdCATH cases allow comparison against a natural reference profile grounded in real MD data; the hand-drawn cases stress-test the pipeline on arbitrary user-defined flexibility patterns with no natural structural prior, which is the relevant regime for genuine de novo design.

Structure preparation. Each FlexFlow-generated backbone was redesigned with ProteinMPNN Dauparas et al. (2022) using the standard co-design protocol (8 sequences per backbone, temperature 0.1, selecting the highest-pLDDT sequence after ESMFold Lin et al. (2023) evaluation). The resulting PDB was cleaned, missing heavy atoms were added, and protonation states were assigned at pH 7 using `gmx pdb2gmx` with the **CHARMM36m** force field Huang et al. (2017) and the TIP3P water model, matching the ATLAS reference parameters. The structure was centred in a cubic box with a 1.0 nm minimum distance to any box edge (`gmx editconf`), solvated with TIP3P water (`gmx solvate`), and neutralised with 0.15 M NaCl (`gmx genion` after a `gmx grompp` ion preprocessing step).

Simulation protocol. The protocol mirrors the ATLAS pipeline as closely as DeltaAI's H100/Grace Hopper hardware permits. Each system passed through four stages: (1) **energy minimisation** via steepest descent until the maximum force fell below 1000 kJ mol^{-1} nm^{-1} (non-bonded forces on GPU, PME on CPU because the steepest-descent integrator does not support GPU PME); (2) **NVT equilibration** for 100 ps at 300 K with a velocity-rescale thermostat ($\tau_T = 0.1$ ps) and position restraints on all heavy atoms (1000 kJ mol^{-1} nm^{-2}); (3) **NPT equilibration** for 100 ps at 300 K and 1 bar with a C-rescale barostat ($\tau_P = 2.0$ ps) and the same position restraints; (4) **production MD** for 100 ns with a 2 fs timestep, LINCS constraints on all bonds involving hydrogen, PME electrostatics (real-space cutoff 1.2 nm, grid spacing 0.12 nm), and a Lennard–Jones cutoff of 1.2 nm with Verlet buffering. For production runs, non-bonded and PME kernels were offloaded to the H100 GPU (`-nb gpu -pme gpu`) with 1 MPI thread and 16 OpenMP threads per GPU (`-ntmpi 1 -ntomp 16 -pin on`), chosen to saturate a single GH200 superchip without contending with other jobs on the node. Frames were written every 100 ps, yielding 1000 frames per replica.

Replica averaging. To reduce sensitivity to initial velocities and to match the ATLAS reference protocol, every system was simulated as **three independent replicas** initialized with different random velocity seeds at the start of NVT equilibration (identical minimised

structure, three different `gen-seed` values in the NVT `.mdp`). The three production trajectories were then processed identically: each trajectory was fit to its average structure using a C_α-only Kabsch alignment (`gmx rmsf -fit`), the per-residue RMSF was computed over the full 100 ns production window of that replica, and the three resulting per-residue RMSF vectors were averaged to produce the final MD-derived RMSF profile used for all comparisons reported in Figure 3 and in the per-protein panels below. This 3-replica-then-average convention matches the ATLAS ground truth against which our Stage 1 and Stage 2 predictors were trained, ensuring that the MD validation is directly comparable to the training signal rather than measuring a different statistical quantity.

Compute footprint. Each replica of a typical 150–250-residue system solvated in a cubic box required approximately 4–8 hours of wall time on a single H100, giving a per-system cost of roughly 12–24 GPU-hours for the three replicas. The full 40-system validation campaign consumed approximately 700 H100 GPU-hours on the DeltaAI `ghx4` partition, submitted as individual single-GPU Slurm jobs (one replica per job, 48-hour wall limit) to maximise throughput across the queue.

Evaluation metric. For each validated design, we report the Pearson correlation r between the 3-replica-averaged MD-derived RMSF and the original target RMSF profile. We additionally report the corresponding S2 prediction (the T5-based evaluator's predicted RMSF correlation) so that readers can compare the learned evaluator's judgement against the physical ground truth on a per-protein basis.

Per-protein profiles. Figures 5, 6, and 7 show the full set of per-residue RMSF profiles for each of the three validation categories. In every panel the black curve is the target RMSF passed to FlexFlow as the conditioning signal, and the coloured curve is the 3-replica-averaged MD-derived RMSF of the ProteinMPNN-redesigned generated backbone; panel titles report the protein identifier, the MD-vs-target Pearson correlation, and the S2 evaluator's prediction. The ATLAS and mdCATH panels show tight agreement between target and MD profiles across most of the sequence, with occasional deviations concentrated at terminal and loop regions where RMSF is intrinsically noisier. The hand-drawn panels reveal the harder regime: several designs track the target profile's qualitative shape (rigid regions remain rigid, flexible regions remain flexible) but compress the dynamic range, reflecting the fact that achieving large absolute RMSF values on de novo backbones with no natural reference structure is substantially harder than reproducing the relative flexibility pattern.

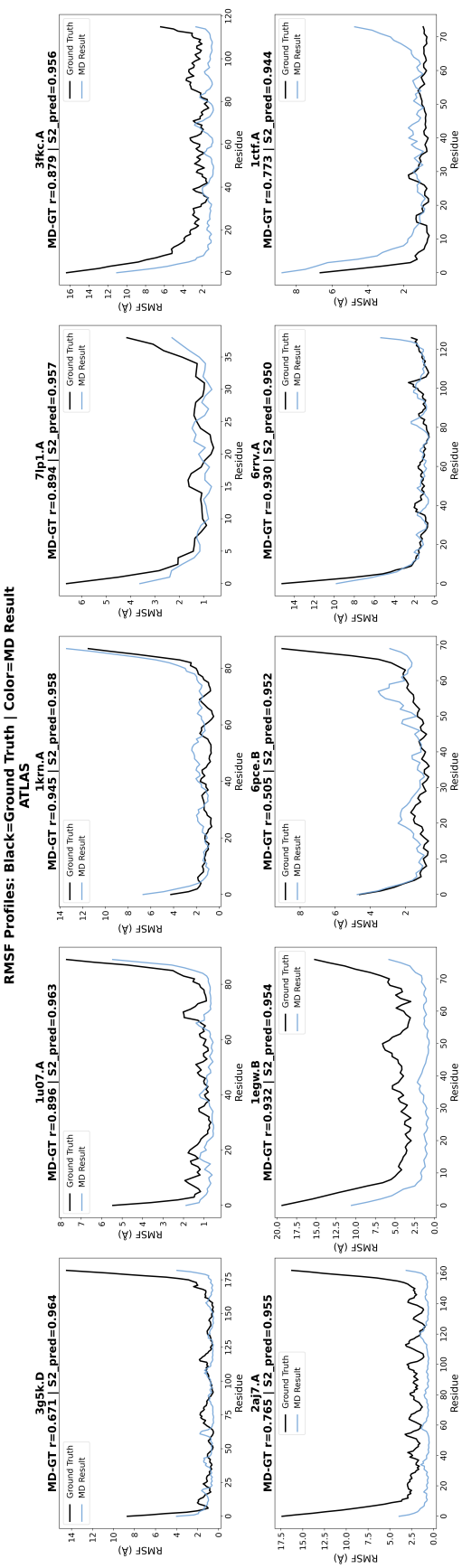


Figure 5: Per-residue RMSF profiles for all 10 ATLAS test proteins. Black: target RMSF; blue: MD-derived RMSF.

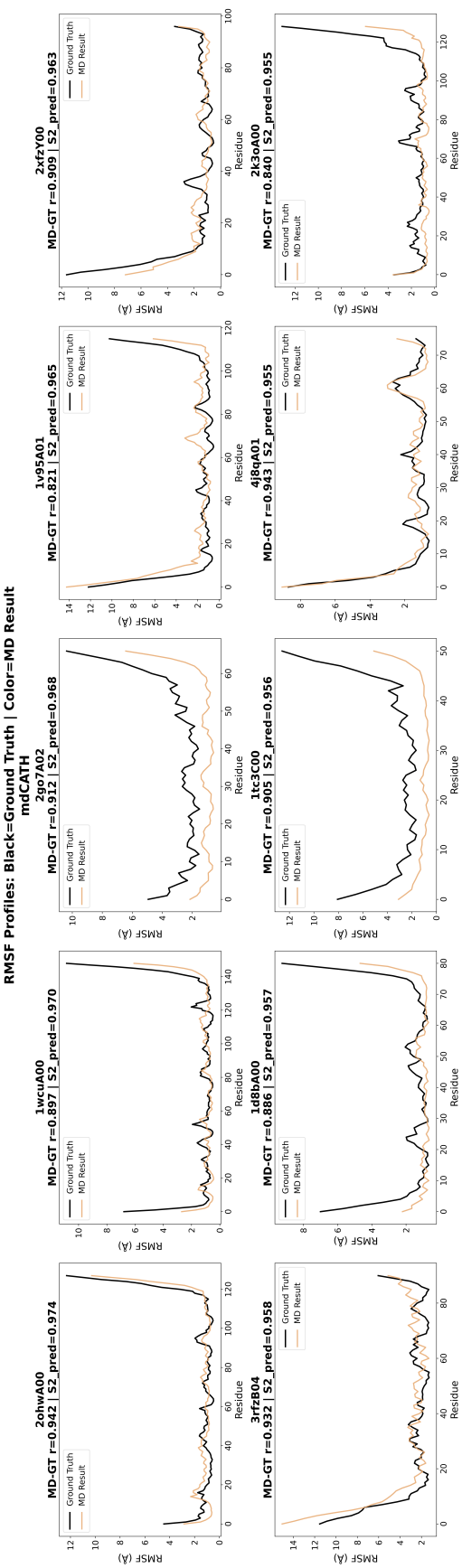


Figure 6: Per-residue RMSF profiles for all 10 mdCATH domains. Black: target RMSF; orange: MD-derived RMSF.

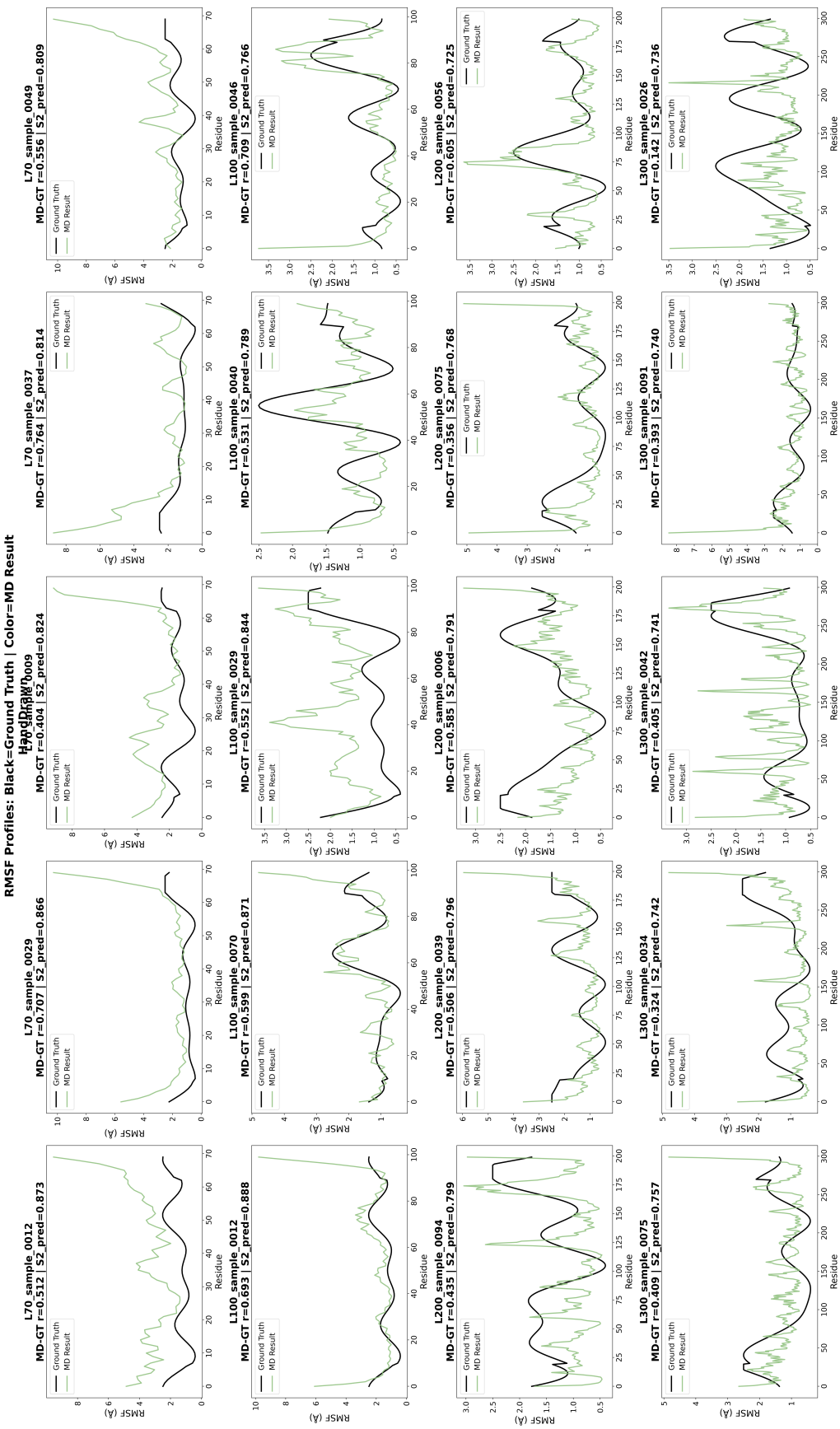


Figure 7: Per-residue RMSF profiles for all 20 hand-drawn flexibility designs. Black: target RMSF; green: MD-derived RMSF.

D Runtime Analysis

We benchmark the generation-only runtime of FlexFlow and FliPS on a single NVIDIA A40 GPU, excluding downstream evaluation steps (ProteinMPNN redesign, ESMFold folding) that are identical for both methods. All FlexFlow experiments use 200 ODE steps with guidance scale $\lambda = 400$.

Table 4: **Generation runtime comparison.** Per-protein generation time (seconds) for FlexFlow (guided and unguided) and FliPS, stratified by protein length. FlexFlow generates a single backbone per protein; FliPS generates 50 samples ($L \leq 300$) or 20 samples ($L > 300$) and selects the top 5 by BackFlip screening. Guidance overhead is the difference between guided and unguided FlexFlow generation time.

	FlexFlow (single sample)				**FliPS** (50 samples per protein)
Length	Guided (sec)	Unguided (sec)	Guidance overhead (sec)	n	Generation (sec)
<100	8.8	6.7	2.1	74	107.2
100–200	10.0	7.5	2.5	90	240.6
200–300	13.9	9.9	4.0	34	374.0
300+	28.1	19.0	9.1	37	415.7
Overall					
ATLAS (mean L=232)	15.8	11.0	4.8	139	416.9
mdCATH (mean L=129)	12.4	10.0	2.4	100	191.8

FlexFlow achieves a 15–26× speedup over FliPS in per-protein generation time. The guidance overhead scales modestly with protein length (2.1 seconds for $L < 100$ to 9.1 seconds for $L > 300$), reflecting the $O(N^2)$ cost of the HierEGT attention layers in the backward pass. Notably, FliPS generates 50 candidate backbones per protein and applies BackFlip screening to select the top 5, whereas FlexFlow produces a single guided backbone directly. Even on a per-sample basis, FlexFlow remains substantially faster (15.8 vs. 83.4 seconds per sample on ATLAS).

E Reproducibility Statement

We will release code, preprocessing scripts, trained checkpoints, and evaluation protocols upon publication. The appendix includes architectural details, hyperparameters, split definitions, and additional ablations.