PROTEO-AA: RESIDUE-AWARE SIDE-CHAIN CO-EVOLUTION FOR ALL-ATOM PROTEIN DESIGN

Anonymous authors
Paper under double-blind review

ABSTRACT

All-atom protein design requires coordination between backbone geometry, amino-acid identity, and side-chain packing. Yet staged design pipelines typically pass information only from backbone generation to sequence and side-chain design, leaving atomic interactions unable to guide the evolving backbone. We introduce ProteoAA, a residue-aware framework that enables bidirectional refinement through trainable adapters connecting frozen backbone and sequence–side-chain models. Backbone residue features inform sequence prediction and side-chain packing, while the resulting all-atom structure provides residue- and pair-level feedback for backbone refinement. A feedback event during diffusion incorporates this atomic information into generation, with side-chain atoms modeled conditionally on residue identity. Across unconditional protein generation, target-conditioned binder design, and side-chain packing benchmarks, Proteo-AA improves sequence–structure consistency, computational binder designability, and atomic geometry. On a standard ten-target binder benchmark, it achieves a mean designability of 31.48%, compared with 21.19% for PXDesign, and ablations support the contributions of both directions of information exchange. These results demonstrate the potential of coupling specialized pretrained models through explicit backbone–side-chain feedback for all-atom protein design.

1 INTRODUCTION

Designing proteins at atomic resolution is a coupled problem across structural scales. Backbone geometry determines the global fold and the arrangement of a functional site, while side-chain conformation determines the local interactions that make that geometry viable: hydrophobic packing, hydrogen-bond and salt-bridge networks, steric compatibility, and atomic complementarity (Alford et al., 2017; Dunbrack Jr & Cohen, 1997). These dependencies are reciprocal. A backbone constrains which residues and rotamers can be accommodated, but the interactions realized by those residues also determine whether the local backbone geometry supports a coherent full-atom structure. Protein design is therefore a refinement process between a global scaffold and its residue-level atomic structure, rather than a sequence of independent backbone, sequence, and packing decisions.

Recent diffusion and flow-matching models have substantially advanced de novo backbone generation, motif scaffolding, and binder design under target conditioning (Trippe et al., 2022; Watson et al., 2023; Team et al., 2025). In many successful systems, however, generation remains staged: a backbone is produced first, an inverse-folding model assigns a sequence, and side chains are subsequently packed or reconstructed. This decomposition provides strong specialized components, but encodes only one direction of the dependency: backbone geometry conditions sequence and side-chain design, whereas packing constraints cannot revise the evolving backbone. At protein interfaces, even a small backbone displacement can change the feasible rotamers and reorganize coupled contacts, yet the staged pipeline provides no route for this atomic information to act on the scaffold that produced it.

All-atom (AA) generative models address this coupling by jointly parameterizing backbone coordinates, residue identities, and side-chain atoms (Kong et al., 2024; Martinkus et al., 2023; Stark et al., 2025; Butcher et al., 2025; Cheng et al., 2026). In these models, coordination among the three variables is learned within a shared generative process. However, a central difficulty in AA diffusion is the discrete–continuous coupling between residue identity and atomic geometry. Unlike small molecules with a fixed atom graph, proteins have residue-dependent atom compositions: each amino

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.

acid type has a distinct side-chain topology and a different number of atoms. Since residue identity is itself part of the design output, the model does not know in advance which side-chain atoms should exist at each position. One common strategy is to represent every residue with a fixed atom template and pad missing positions with virtual or ghost atoms (Stark et al., 2025; Cheng et al., 2026). While convenient, this strategy introduces nonphysical degrees of freedom and raises ambiguous questions about how nonexistent atoms should be generated, supervised, and interpreted. Another strategy is to use latent diffusion models (Kong et al., 2025; 2026; Geffner et al., 2026), which avoid explicit diffusion over variable atom sets by generating full-atom structures through a learned latent space and decoder. However, this shifts the burden of full-atom accuracy to the latent representation and reconstruction network, making fine-grained side-chain geometry, rotameric fidelity, and controllable backbone–side-chain interactions dependent on the expressiveness of the latent space.

These limitations suggest that AA protein design should not be formulated as coordinate diffusion over a larger atom set. Instead, the model must explicitly address the fact that side-chain coordinates are only well-defined after residue identity is specified, while residue identity itself should be informed by the side-chain interactions it enables. This creates a chicken-and-egg problem: *backbone geometry and residue identity determine which side-chain atoms exist, but side-chain packing and interaction geometry should in turn refine the backbone and residue identity*. Making this two-way interaction effective requires addressing three challenges. First, backbone diffusion and all-atom design use different representations, so the sequence and side-chain geometry must be converted into feedback that the backbone denoiser can use. Second, this feedback must be incorporated at the correct sampling step: side chains built on an intermediate backbone prediction should refine that prediction at the same noise level, before the diffusion solver advances. Third, the final sequence must reflect the updated backbone. Reusing a sequence selected before the backbone was refined would leave the final residue choices unaffected by the feedback.

To this end, we introduce **Proteo-AA**, a residue-aware framework for bidirectional backbone–side-chain refinement. Its backbone-to-side-chain interface provides per-residue denoiser features for sequence and side-chain design, while its side-chain-to-backbone interface converts the resulting packed structure into residue- and pair-level feedback. We instantiate these interfaces using a PXDesign backbone denoiser (Team et al., 2025) and a FAMPNN sequence–side-chain model (Shuai et al., 2025). At inference, Proteo-AA performs a single feedback event at a predefined noise level. PXDesign first predicts a provisional clean backbone, which FAMPNN uses together with the denoiser's residue features to propose a sequence and pack its side chains. The resulting all-atom structure provides feedback for a second PXDesign evaluation at the same noisy state and noise level, yielding a corrected backbone prediction. Following feedback, the backbone diffusion solver continues until backbone generation is complete. FAMPNN then uses residue features and the completed backbone to generate the final sequence and side-chain conformations.

We design controlled experiments to test whether improvements come from side-chain information. An additional denoiser call could improve the backbone even without this information, and decoding the sequence again could produce a better result simply through random sampling. To separate these effects, we start each comparison from the same noisy backbone state and use the same sequence re-decoding procedure across conditions. We then compare all-atom feedback removing backbone-to-side-chain conditioning and side-chain-to-backbone feedback respectively, using the same compute budget. These comparisons show that side-chain packing provides benefits beyond extra denoising and sequence re-sampling.

Our contributions are summarized as follows:

- We formulate AA protein design as reciprocal refinement between global backbone geometry and residue-level atomic interactions, and identify the missing reverse path in staged design pipelines.
- We introduce a residue-aware interface that translates between the heterogeneous states of backbone diffusion and sequence–side-chain design, enabling information to flow in both directions.
- We develop a state-aligned refinement procedure that performs provisional AA design, corrective backbone evaluation, post-feedback sequence re-decoding, and final side-chain generation.
- We construct matched controls and side-chain-aware evaluation that separate atomic feedback from additional denoising and stochastic re-sampling.

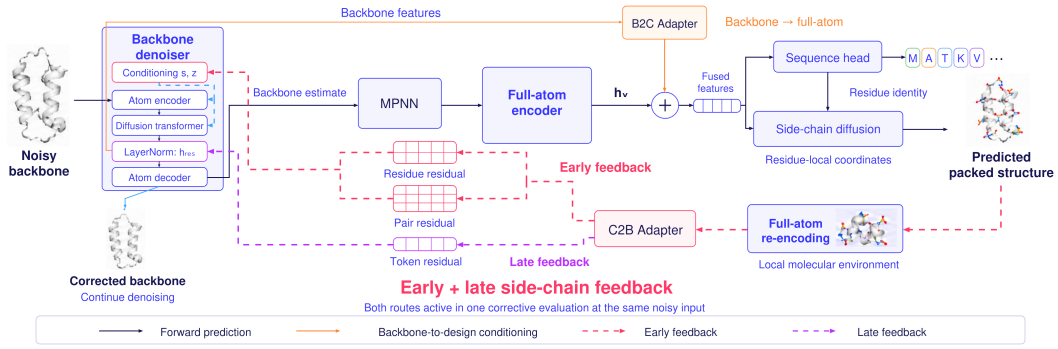

Figure 1: **Bidirectional residual coupling in Proteo-AA.** The backbone denoiser produces a provisional clean-backbone estimate and residue features $\mathbf{h}_{\text{res}}$. B2C Adapter maps these features to a noise-conditioned residual, which is added to the full-atom MPNN representations $\mathbf{h}_V$ to condition both sequence prediction and side-chain diffusion. The resulting packed structure provides feedback through two branches of C2B Adapter: the early branch updates the residue and pair conditioning tensors (s, z), while the late branch updates the normalized residue tokens immediately before coordinate decoding. Orange arrows indicate backbone-to-design conditioning; red and purple dashed arrows indicate early and late feedback, respectively.

2 METHOD

2.1 PRELIMINARIES

Full-atom protein representation. A protein of length L is represented by an amino-acid sequence $a = (a_1, \ldots, a_L)$, where $a_i \in \mathcal{V}$ and $\mathcal{V}$ contains the 20 standard amino acids. Each residue has backbone coordinates $\mathbf{x}_i^{\text{bb}} = (\mathbf{x}_i^N, \mathbf{x}_i^{C_\alpha}, \mathbf{x}_i^C, \mathbf{x}_i^O) \in \mathbb{R}^{4\times 3}$. Its side-chain atom set, denoted by $\mathcal{A}_{\text{sc}}(a_i)$, depends on the residue identity. We write its side-chain coordinates as $\mathbf{x}_i^{\text{sc}} = (\mathbf{x}_{ij}^{\text{sc}})_{j\in\mathcal{A}_{\text{sc}}(a_i)}$ and the full-atom structure as $\mathbf{x} = (\mathbf{x}^{\text{bb}}, \mathbf{x}^{\text{sc}})$. Because this atom composition is residue-dependent, we track which side-chain atoms are physically present with a validity mask $\mathcal{A}$; accommodating this variability is a central challenge for full-atom design. For binder design, $\mathcal{T}$ denotes the fixed target context and $\mathbf{M}_B$ indicates which binder residues are designable.

Backbone and side-chain diffusion states. At backbone diffusion step t, the noisy binder backbone is denoted by $\mathbf{x}_t^{\text{bb}}$, with $\mathbf{x}_t^{\text{bb}} = \mathbf{x}_0^{\text{bb}} + \sigma_t \boldsymbol{\epsilon}$ and $\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$, where σ_t is the noise level presented to the denoiser. The backbone module predicts clean coordinates $\hat{\mathbf{x}}_0^{\text{bb}}$. The side-chain module uses a separate internal diffusion process indexed by τ. The internal index τ is distinct from the backbone step t: side-chain diffusion is run within each full-atom design operation, rather than maintained as a state of the outer backbone trajectory.

Problem objective. Our goal is to learn a generative model $p(\mathbf{x}^{\text{bb}}, a, \mathbf{x}^{\text{sc}}|\mathcal{T})$ that jointly produces backbone coordinates, amino acid identities, and residue-specific side-chain coordinates.

2.2 OVERVIEW OF PROTEO-AA

The architecture is summarized in Figure 1. The backbone module $B_{\boldsymbol{\theta}}$ and the side-chain module $S_{\boldsymbol{\phi}}$ are coupled through two residual adapters to exchange information. B2C Adapter provides $S_{\boldsymbol{\phi}}$ with access to the internal representations of the backbone denoiser, rather than limiting it to the predicted backbone structure alone. C2B Adapter feeds the generated all-atom prediction back into the backbone module, allowing the backbone prediction to be refined in light of the resolved side-chain configurations. The output projection layers of both adapters are initialized to zero, so the coupled model recovers the behavior of the two components when training starts.

The backbone module is built upon PXDesign (Team et al., 2025). Given $(\mathbf{x}_t^{\text{bb}}, \sigma_t; \mathcal{T})$, it produces a denoised backbone prediction $\hat{\mathbf{x}}_0^{\text{bb}}$ together with residue-level features $\mathbf{h}_{\text{res}} \in \mathbb{R}^{\psi_{\text{res}}}$, where the latter are extracted from the normalized token representations immediately before the atom-level decoder.

At this stage, the hidden state remains residue-level and thus directly captures the model's internal representation at each residue before it is decoded into atomic coordinates. The denoised backbone $\hat{\mathbf{x}}_0^{\mathrm{bb}}$ is provided to $S\phi$ as structural input, while $\mathbf{h}_{\mathrm{res}}$ is passed to B2C Adapter for further processing.

Inspired by FAMPNN (Shuai et al., 2025), which highlights the benefit of incorporating side-chain structural context into sequence design, the side-chain module $S_{\boldsymbol{\phi}}$ represents each residue jointly by its amino-acid identity and side-chain conformation. Given current backbone geometry and the available sequence a and side-chain context $\mathcal{C}$, the all-atom MPNN produces residue-level representations $\mathbf{h}_V \in \mathbb{R}^{\psi_{\mathrm{MPNN}}}$, which are shared by the pretrained sequence prediction head and side-chain diffusion module. We formulate the joint distribution of residue identity and side-chain conformation as

$$p_\phi\left(a_i, \mathbf{x}_i^{\mathrm{sc}} \mid \mathcal{C}\right) = p_\phi\left(a_i \mid \mathcal{C}\right) p_\phi\left(\mathbf{x}_i^{\mathrm{sc}} \mid a_i, \mathcal{C}\right). \tag{1}$$

Accordingly, side-chain generation is conditioned on the predicted residue identity $\hat{a}_i$ during inference and on the ground-truth residue identity during training. Inference proceeds through iterative unmasking, progressively resolving residue identities and generating their corresponding side-chain conformations. The resolved residue identities, together with confidence-filtered side-chain predictions, are incorporated into the context for subsequent iterations.

2.3 Bidirectional Residual Coupling

We link two modules in two directions. B2C Adapter improves the predicted sequence and side chains given a noisy backbone, whereas C2B Adapter improves the backbone given a packed structure.

2.3.1 Backbone–to–side-chain (B2C) Adapter

B2C Adapter $A_{\boldsymbol{\vartheta}} : \mathbb{R}^{\psi_{\mathrm{res}}} \to \mathbb{R}^{\psi_{\mathrm{MPNN}}}$ establishes the information feedback from the backbone denoiser $B_{\boldsymbol{\theta}}$ to the side-chain module $S_{\boldsymbol{\phi}}$. Specifically, the full-atom MPNN first encodes the current backbone geometry and available sequence and side-chain context into residue-level representations $\mathbf{h}_V$. B2C Adapter then maps the backbone hidden states $\mathbf{h}_{\mathrm{res}}$ into the MPNN feature space and injects them as a residual after the full-atom MPNN:

$$\mathbf{h}_V^{\mathrm{fused}} = \mathbf{h}_V + \mathbf{M}_B \odot A_{\boldsymbol{\vartheta}}\left(\mathbf{h}_{\mathrm{res}}, \sigma_t\right), \tag{2}$$

where $\mathbf{M}_B$ specifies the mutable positions at which the residual is applied. The fused representations $\mathbf{h}_V^{\mathrm{fused}} \in \mathbb{R}^{\psi_{\mathrm{MPNN}}}$ are subsequently passed to both the pretrained sequence prediction head and the side-chain diffusion module. Besides, the B2C Adapter is conditioned on the noise level σ_t, allowing the injected information to adapt to the current stage of the backbone denoising trajectory.

2.3.2 Sidechain-to-back-bone (C2B) Adapter

After all-atom generation, we construct a packed state $\mathcal{P}$, containing the predicted backbone $\hat{\mathbf{x}}^{\mathrm{bb}}$, sequence a, and side-chain conformations $\hat{\mathbf{x}}^{\mathrm{sc}}$, together with atom-validity and confidence information. C2B Adapter processes $\mathcal{P}$ through two independently parameterized neural networks, $E_{\boldsymbol{\xi}^e}$ and $E_{\boldsymbol{\xi}^l}$, whose outputs are injected at different stages of the backbone denoiser. We refer to these networks collectively as $E_{\boldsymbol{\xi}}$ when the distinction between the two injection pathways is not relevant.

Early conditioning. The early branch of the C2B Adapter, $E_{\boldsymbol{\xi}^e}$, uses the generated side-chain structure to refine the backbone denoiser's conditioning. Specifically, it adds learned corrections to the residue-level features $\mathbf{s}$ and pair-level features $\mathbf{z}$ produced by the pretrained conditioning stack.

These corrections are derived from the packed state $\mathcal{P}$, which contains the predicted backbone $\hat{\mathbf{x}}^{\mathrm{bb}}$, sequence a, and side-chain conformations $\hat{\mathbf{x}}^{\mathrm{sc}}$, together with atom-validity and confidence information. The atom-validity mask $\mathcal{A}$ is updated after packing to reflect the atoms present in each generated residue. This branch does not use the full-atom MPNN representations $\mathbf{h}_V$; its feedback is constructed from the explicit packed structure and its associated attributes.

To construct the corrections, $E_{\boldsymbol{\xi}^e}$ first encodes $\mathcal{P}$ into two intermediate representations: a residue feature $\mathbf{U}_i$ describing the identity, side-chain geometry, and prediction uncertainty of residue i, and a directed pair feature $\mathbf{V}_{ij}$ describing the relative geometry and side-chain interactions between spatially neighboring residues i and j. Their complete definitions are provided in App. B.1. Two

zero-initialized output heads, conditioned on σ_t, map $\mathbf{U}_i$ and $\mathbf{V}_{ij}$ to residue-level corrections $\Delta\mathbf{s}_i$ and pair-level corrections $\Delta\mathbf{z}_{ij}$, respectively. The complete conditioning updates are written as

$$(\Delta\mathbf{s}, \Delta\mathbf{z}) = E_{\boldsymbol{\xi}^e}(\mathcal{P}, \sigma_t), \qquad \mathbf{s}' = \mathbf{s} + \mathbf{M}_B \odot \Delta\mathbf{s}, \qquad \mathbf{z}' = \mathbf{z} + \mathbf{M}_{BB} \odot \Delta\mathbf{z}. \tag{3}$$

Here, $\mathbf{M}_B$ restricts residue updates to the designed region, while $\mathbf{M}_{BB}$ permits pair updates only when both residues belong to that region. The denoiser then uses $(\mathbf{s}', \mathbf{z}')$ for backbone refinement, leaving the pretrained conditioning stack unchanged.

The updates are complementary: $\mathbf{s}'$ supplies residue-level conditioning to every diffusion block, while $\mathbf{z}'$ supplies pairwise attention biases that describe relationships between specific residues. This pair-level pathway allows the denoiser to account for side-chain contacts when refining the backbone. We call this conditioning *early* because it is injected before the denoiser's subsequent processing, allowing the feedback to propagate across residues. The exact injection sites are detailed in App. B.1.

Late adaptation. The late branch of the C2B Adapter, $E_{\boldsymbol{\xi}^l}$, provides an additional correction to the backbone denoiser's residue tokens immediately before coordinate decoding. During the corrective pass, the denoiser first processes the updated conditioning $(\mathbf{s}', \mathbf{z}')$ to produce $\widetilde{\mathbf{h}}_{\text{res}}$, the normalized residue tokens before the coordinate decoder.

To construct the late correction, we re-encode the generated full-atom structure in $\mathcal{P}$ using the full-atom MPNN. $E_{\boldsymbol{\xi}^l}$ uses the resulting residue representations, together with additional packed-state features detailed in App. B.1, to predict a token-level residual. Unlike $E_{\boldsymbol{\xi}^e}$, it therefore has access to learned MPNN representations of the generated structure. The residual is computed as

$$\Delta\mathbf{h}_{\text{res}} = E_{\boldsymbol{\xi}^l}(\mathcal{P}, \sigma_t), \qquad \mathbf{h}'_{\text{res}} = \widetilde{\mathbf{h}}_{\text{res}} + \mathbf{M}_B \odot \Delta\mathbf{h}_{\text{res}}. \tag{4}$$

The coordinate decoder then uses $\mathbf{h}'_{\text{res}}$ to predict the refined backbone.

Both branches operate within the same corrective denoiser pass but introduce feedback at different stages. Early feedback enters before global cross-residue processing, allowing side-chain information to influence interactions across residues. Late feedback enters after this global processing and directly adjusts the tokens used for coordinate decoding. Its influence on other residues is therefore limited to the atom decoder's local attention windows. Together, the two branches provide side-chain information both during global backbone processing and at the final coordinate-decoding stage.

2.4 Light-weighted Training

Both base modules $B_{\boldsymbol{\theta}}$ and $S_{\boldsymbol{\phi}}$ remain frozen. The two adapters are first trained separately. Since the quality of C2B feedback depends on the packed structure produced by B2C, we subsequently applied jointly training to capture their interaction and align training with inference workflow.

B2C Adapter. Training examples are structures whose designed backbone region is corrupted to a noise level σ_t, and the backbone features $\mathbf{h}_{\text{res}}$ of equation 2 are detached, so only $A_{\boldsymbol{\vartheta}}$ is updated and gradients reach it through the frozen sequence head and side-chain denoiser that read h'. The objective is $\mathcal{L}_{\text{design}}(\boldsymbol{\vartheta}) = \lambda_{\text{aa}}\mathcal{L}_{\text{aa}} + \lambda_{\text{sc}}\mathcal{L}_{\text{sc}}$, combining cross-entropy over masked binder identities with the donor side-chain diffusion objective with EDM weighting in residue-local frames. Masking conventions, loss reductions, and the calibration of λ_{aa} are given in App. B.2.

We train the adapter at a fixed noise level, motivated by an empirical sweep over the noise schedule on held-out structures. The sweep reveals a clear dependence on backbone corruption: adaptation provides larger gains at higher noise levels, whereas its benefit diminishes and can become slightly negative as the backbone approaches the low-noise end of the schedule (App. B.4). We therefore choose $\sigma_t = 0.429$ as a practical operating point at which adaptation still yields a positive gain while the backbone is sufficiently denoised to provide a reliable structural context for side-chain prediction. Applying feedback at this relatively late stage also limits the amount of subsequent denoising required after all-atom generation. This choice therefore balances the quality of the side-chain context used for feedback, the remaining opportunity to refine the backbone, and the additional inference cost introduced by the feedback step.

C2B Adapter. For individual C2B Adapter training, B2C Adapter is bypassed, and the sequence is held fixed, so the feedback state is generated solely by the frozen pretrained modules. The provisional

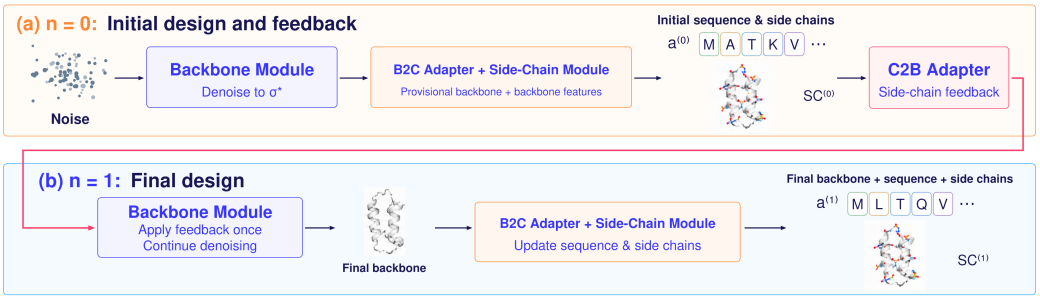

Figure 2: **Single-event full-atom feedback during inference.** (a) At the backbone sampling step whose realized noise level is closest to $\sigma^\star$, B2C Adapter and the side-chain module use the provisional clean-backbone estimate and backbone features to generate an intermediate sequence and side-chain packing. C2B Adapter converts the resulting packed state into early and late feedback residuals. (b) The backbone denoiser is re-evaluated once at the same noisy state and noise level with both feedback branches active. The corrected prediction is used to continue backbone sampling without further feedback. After backbone denoising is complete, the side-chain module generates the final sequence and side-chain conformations on the final backbone and cached B2C conditioning residuals. The labels $n = 0$ and $n = 1$ denote the intermediate and final full-atom design stages, respectively.

backbone, packed side chains, and feedback inputs are precomputed without gradient tracking and reused throughout training, leaving only $E_{\boldsymbol{\xi}}$ trainable. The corrected backbone is supervised against:

$$\mathcal{L}_{\mathrm{bb}}^{\mathrm{post}}(\boldsymbol{\xi}) = w_{\mathrm{bb}}(\sigma_t)\,\frac{\sum_n r_n \left\|\hat{\mathbf{x}}_{n,\mathrm{post}}^{\mathrm{bb}} - \mathbf{x}_n^{\mathrm{bb,GT}}\right\|_2^2}{\sum_n r_n}, \tag{5}$$

where r_n selects resolved backbone atoms in the designed region $\mathbf{M}_B$ and w_{bb} is the EDM reconstruction weight. Unlike B2C Adapter, C2B Adapter is trained with σ_t sampled per example from a predefined noise window. The gate in Eq. equation 3 is fixed to one within this window and tapers outside it, limiting feedback at unseen noise levels. We keep this gate fixed because jointly learning it with a zero-initialized projection would introduce a stationary point at initialization. The corrective pass uses the same noisy state and noise level as the provisional pass, with feedback as the only difference, and predictions are compared without rigid alignment. Since both injection paths are zero-initialized, the untrained adapter exactly recovers the provisional prediction.

2.5 Feedback at Inference

The sampling procedure is summarized in Figure 2. Backbone denoising proceeds normally until the solver reaches the step whose realized noise level is closest to a chosen level $\sigma^\star$, where the realized noise level is the level presented to the denoiser after sampler churn (App. B.3). At this step, $B_{\boldsymbol{\theta}}$ produces a provisional clean-backbone estimate together with $\mathbf{h}_{\mathrm{res}}$, and B2C Adapter conditions $S_{\boldsymbol{\phi}}$ to generate an intermediate sequence and side-chain packing on that estimate. C2B adapter converts the predicted state into feedback residuals, which are immediately applied in a second backbone-denoiser evaluation at the same realized noise level. Within C2B Adapter, the early branch updates the residue and pair conditioning tensors, while the late branch updates the normalized residue tokens immediately before coordinate decoding. Sampling then continues along the original noise schedule without further feedback. Notably, the intermediate sequence and side chains are used only to construct the feedback, while the final sequence and side chains are generated anew from the fully denoised backbone using cached B2C adapter and $S_{\boldsymbol{\phi}}$. As a consequence, our coupled sampler adds one corrective backbone-denoiser evaluation and one intermediate call to $S_{\boldsymbol{\phi}}$, together with lightweight adapters, which avoids adding overhead computation burden.

3 Experiments

To demonstrate the effectiveness of Proteo-AA for all-atom protein modeling, we conduct extensive experiments spanning unconditional protein generation, conditional binder design, and conditional

Table 1: **Proteo-AA achieves SOTA results on unconditional all-atom design, for lengths between 100 and 500 residues.** Diversity, novelty, and secondary structure computed on all-atom co-designable samples. Best results are marked in bold.

Model	Codesignability (%) ↑		Designability (%) ↑		Sec. Struct. (%)		Diversity ↑		Novelty ↓
	All-Atom	Cα-Only	PMPNN@1	PMPNN@8	α	β	Str	Seq	PDB
A-CODE (Cheng et al., 2026)	81.2	87.0	91.6	97.0	50	14	70	252	–
P(all-atom) (Qu et al., 2024)	36.7	37.9	44.4	57.9	56	17	134	148	**0.72**
APM (Chen et al., 2025)	19.0	32.2	42.8	61.8	73	8	32	64	0.84
PLAID (Lu et al., 2024)	11.0	19.2	23.8	37.6	44	14	25	38	0.89
ProteinGenerator (Lisanza et al., 2025)	9.8	17.8	42.8	54.2	78	5	12	28	0.83
MultiFlow (Campbell et al., 2024)	–	73.2	81.4	94.0	72	12	**209**	247	–
Protpardelle (Chu et al., 2024)	8.8	35.2	43.8	56.2	65	14	10	37	0.79
La-Proteina (Geffner et al., 2026)	68.4	72.2	82.6	93.8	72	5	206	216	0.75
La-Proteina (w/ tri)	75.0	78.2	84.6	94.6	73	6	129	252	0.82
Proteo-AA	**82.2**	**89.4**	**92.6**	**99.0**	44	15	96	**267**	0.80

binder side-chain packing. Together, these evaluations assess Proteo-AA across complementary aspects of protein generation, from target-conditioned binder generation to fine-grained recovery of side-chain conformations and interface packing.

We evaluate all methods under matched input conditions and, where applicable, comparable inference budgets. In particular, we match FAMPNN and Proteo-AA by assigning the same number of solver calls and decoding steps to reflect authenticated modeling quality.

3.1 Unconditional Protein Generation

Evaluation setup. Following La-Proteina (Geffner et al., 2026) and A-CODE(Cheng et al., 2026), we generate 100 unconditional samples for each sequence length $L \in \{100, 200, 300, 400, 500\}$. We evaluate codesignability using the model-generated sequences and designability using sequences redesigned by ProteinMPNN, measuring structural self-consistency against the generated structures in both cases. Codesignability is reported using both all-atom and C_α-only structural comparisons. For designability, PMPNN@1 and PMPNN@8 denote evaluation with one and eight redesigned sequences per backbone, respectively, with the best-matching prediction used for PMPNN@8. We additionally report secondary-structure composition, structural and sequence diversity, and similarity to known structures in the PDB as a measure of novelty. Inference time is measured in seconds per sample on a single NVIDIA A100-80G GPU.

Generation quality. As shown in Tab. 1, Proteo-AA achieves state-of-the-art codesignability and designability among the evaluated methods, outperforming A-CODE, La-Proteina variants, and other baselines. It attains 82.2% all-atom and 89.4% C_α-only codesignability, demonstrating strong consistency between generated sequences and structures at both atomic and backbone levels. After sequence redesign with ProteinMPNN, Proteo-AA achieves 92.6% designability with PMPNN@1 and 99.0% with PMPNN@8, surpassing strong baselines including MultiFlow and La-Proteina. These results support the quality of both the generated sequence–structure pairs and the backbones evaluated independently of their original sequences. Inference speed comparison is in App. D

3.2 Conditional binder evaluation benchmark

Experimental setup. We evaluate Proteo-AA on the same binder-design benchmark as AlphaProteo (Zambaldi et al., 2024), which comprises 10 challenging targets with diverse structural characteristics. For each target, its sequence and structure are provided, as well as binding-site hotspots if specified. We generate 328–728 candidate binders per target, with lengths ranging from 80 to 130 residues. Following the PXDesign (Team et al., 2025) and ProtDBench (Liu et al., 2026) evaluation protocol, we assess candidates using the AF2-IG (Watson et al., 2023) filter and define designability as the percentage of candidates that pass all filtering criteria. Tab. 2 reports per-target designability and its unweighted mean across all targets, together with ablations of the B2C and C2B adapters and their combination.

Table 2: **Proteo-AA and its ablations attain the best performance on 6 of 10 targets and the highest average designability on the conditional binder design benchmark.** Proteo-AA achieves a 48.6% relative improvement over the strongest baseline, PXDesign. We report the percentage of designable samples for each target.

Method	BHRF1	H1	IL17A	IL7RA	IR	PDL1	SC2RBD	TNFa	TrkA	VEGFA	Mean
Protpardelle-1c (Lu et al., 2025)	3.73	0.27	0.00	0.09	0.19	7.04	0.99	0.00	3.52	0.17	1.60
BoltzGen (Stark et al., 2025)	14.56	19.21	0.31	12.41	22.34	14.26	0.69	0.88	**28.65**	7.69	12.10
ODesign (Zhang et al., 2025)	42.23	7.29	0.10	7.61	16.82	10.26	3.13	0.00	9.99	3.71	10.11
RFDiffusion-3 (Butcher et al., 2025)	33.38	0.89	0.82	6.47	18.17	16.81	4.34	0.00	14.44	2.95	9.83
PXDesign (Team et al., 2025)	43.90	12.08	0.82	29.80	25.04	45.33	11.20	3.43	23.55	16.72	21.19
A-CODE (Co-Design) (Cheng et al., 2026)	22.87	55.71	1.24	4.05	41.67	28.70	**37.50**	**7.57**	3.70	0.96	20.40
A-CODE (PMPNN) (Cheng et al., 2026)	25.00	**65.59**	1.79	4.93	30.09	39.96	28.05	6.16	6.87	1.37	20.98
Proteo-AA	**80.18**	12.50	**4.12**	**53.17**	46.91	**66.73**	9.45	5.28	10.39	**26.10**	**31.48**
w/o B2C adapter	79.27	10.49	0.00	52.11	**47.84**	62.50	10.37	6.34	10.39	16.62	29.59
w/o C2B adapter	66.77	6.17	1.10	39.61	43.83	60.39	4.27	4.23	12.50	12.50	25.14
w/o both	66.77	6.17	**4.12**	39.61	41.67	62.50	2.13	2.11	6.34	10.44	24.19

Binder design performance. In Tab. 2, Proteo-AA achieves the highest mean designability of 31.48% across all targets, exceeding PXDesign and A-CODE (PMPNN) by 10.29% and 10.50%, respectively. It obtains the best results on six targets: BHRF1, IL17A, IL7RA, IR, PDL1, and VEGFA. Performance remains target-dependent, with A-CODE leading on H1, SC2RBD, and TNFa, and BoltzGen on TrkA. These demonstrate strong performance of our Proteo-AA.

Effect of side-chain-to-backbone feedback. We ablate the B2C adapter, the C2B adapter, or both. To distinguish the benefit of bidirectional information exchange from that of additional denoising or sequence sampling, we keep the denoising and sequence-sampling schedule unchanged. As shown in Tab. 2, the full model consistently outperforms the ablated variants, with the largest degradation occurring when feedback pathways are removed.

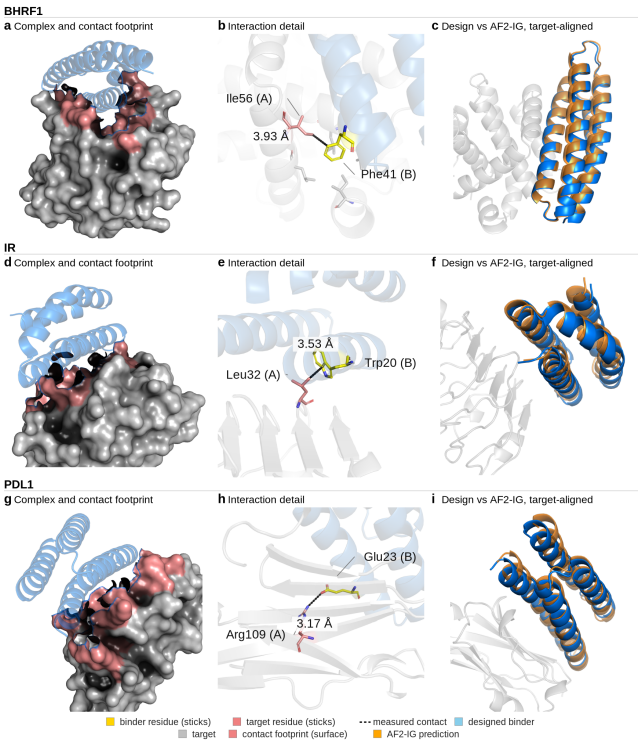

Figure 3: Representative binders for BHRF1, IR, and PDL1. Left panels (a,d,g) show the designed binder–target complexes and target contact footprints; middle panels (b,e,h) highlight representative residue-level interface interactions and their interatomic distances; right panels (c,f,i) compare the designed binder structures (blue) with AF2-IG predictions (orange) after alignment. The close structural overlap and high retention of designed interface contacts indicate recovery of the intended binding modes under independent structure prediction.

Case-study and visualization. We further examine three representative binder designs targeting BHRF1, IR, and PDL1 to assess whether interface geometry is preserved under independent structure prediction. As shown in Fig. 3, designs form broad contact footprints on their respective targets and exhibit specific, chemically plausible residue-level interactions. Specifically, we have BHRF1 Ile56–Phe41 (3.93 Å), IR Leu32–Trp20 (3.53 Å), and a PDL1 Arg109–Glu23 electrostatic interaction (3.17 Å). After alignment on the target, AF2-IG predictions closely recover the designed binder poses, with pose RMSDs of 0.89, 1.78, and 1.27 Å for BHRF1, IR, and PDL1. Our designs also retain 93%, 85%, and 92% of the designed interface contacts.

Table 3: **Proteo-AA improves side-chain packing while providing backbones that better support side-chain reconstruction.** Proteo-AA achieves the best performance on six of seven reported metrics under both backbone settings. Moreover, using Proteo-AA corrected backbones improves all six reconstruction metrics relative to PXDesign backbones for every evaluated packing method. Bold indicates the best result within each backbone setting.

Method	RMSD ↓	χ-Rec@20 ↑	Rot. Rec. ↑	χ_1@20 ↑	SC–SC ↑	SC–Env ↑	Bad Bond ↓
PXDesign backbone							
FAMPNN (Shuai et al., 2025)	2.1959	0.6207	**0.6357**	0.7075	0.7967	0.8339	0.0757
LigandMPNN (Dauparas et al., 2025)	2.2813	0.6190	0.6196	0.6916	0.7720	0.8143	0.0735
PackPPI (Zhang & Xiong, 2025)	2.3033	0.5889	0.5936	0.6647	0.7607	0.8024	0.0734
Proteo-AA	**1.5207**	**0.6355**	0.6246	**0.7179**	**0.8209**	**0.8611**	**0.0009**
Proteo-AA–corrected backbone							
FAMPNN (Shuai et al., 2025)	1.4554	0.6545	**0.6754**	0.7474	0.8228	0.8624	0.0540
LigandMPNN (Dauparas et al., 2025)	1.5597	0.6490	0.6434	0.7242	0.7954	0.8449	0.0504
PackPPI (Zhang & Xiong, 2025)	1.6086	0.6173	0.6184	0.6982	0.7856	0.8324	0.0530
Proteo-AA	**1.4331**	**0.6652**	0.6628	**0.7547**	**0.8294**	**0.8724**	**0.0502**

3.3 Conditional Binder Design Side Chain Packing

Binder design requires not only compatible backbone geometry but also well-packed side chains that support favorable interactions at the interface. We therefore evaluate side-chain packing in conditional binder design, using the target structure as context and introducing controlled levels of structural corruption into the binder backbone to assess packing quality under backbone uncertainty.

Inference protocol. For each binder-target complex, we keep the target structure fixed and evaluate side-chain packing on both PXDesign-generated and Proteo-AA corrected backbones. For backbone correction, we corrupt the binder backbone at five noise levels, corresponding to five points in the backbone module's EDM noise schedule. The backbone module then denoises the corrupted binder conditioned on the clean target context. All side-chain packing methods receive the same binder backbone, native binder sequence, and a clean, fixed target; only binder side chains are predicted and scored. We additionally report all evaluation metrics and whole-complex packing results in App. C, where both the receptor and binder are noised and scored.

Baseline construction. We compare against LigandMPNN (Dauparas et al., 2025) and PackPPI (Zhang & Xiong, 2025), both designed for interface or binder-oriented side-chain modeling. To enable a fairer comparison with FAMPNN (Shuai et al., 2025), we additionally fine-tune FAMPNN using the official code on the same training data as our side-chain module. This controls for differences in training-set composition and allows the comparison to more directly reflect differences in model architecture and side-chain modeling capability.

Side-chain evaluation. On the binder-only side-chain packing task (Tab. 3), Proteo-AA achieves the best performance on six of seven reported metrics under both backbone settings. FAMPNN retains the highest rotamer recovery. The largest gains over competing methods occur on PXDesign backbones, where Proteo-AA corrected backbones improve all six reported reconstruction metrics for every method. These results suggest that our coupled approach improves backbone-side-chain compatibility, providing a more favorable structural context for side-chain packing.

4 Conclusion

We introduced ProteoAA, a residue-aware framework that couples backbone generation with sequence and side-chain design through bidirectional adapters. By allowing packed side chains to inform backbone refinement, ProteoAA incorporates atomic interactions into the generation process while keeping the pretrained components frozen. Experiments show improved sequence–structure consistency, binder designability, and side-chain packing quality, with ablations supporting the benefits of bidirectional coupling. These results highlight reciprocal backbone–side-chain refinement as a practical approach to more coherent all-atom protein design.

AI USE STATEMENT

In this work, we used generative AI tools for assisting with code writing for experiments, literature review, and writing refinement. We have not used generative AI tools for developing methods, formulating research ideas, proposing hypotheses, or determining the main scientific arguments and conclusions, and any other required disclosure task is not applicable to this work. Additionally, we used generative AI tools for plot generation, spelling checks, and language polishing. We have reviewed all AI-assisted work. We verified and tested all AI-assisted code for correctness, reviewed AI-assisted literature searches to ensure that the cited literature was accurate and traceable to the original sources, and manually reviewed all AI-assisted text and plots for accuracy and consistency with the underlying data. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Rebecca F Alford, Andrew Leaver-Fay, Jeliazko R Jeliazkov, Matthew J O'Meara, Frank P DiMaio, Hahnbeom Park, Maxim V Shapovalov, P Douglas Renfrew, Vikram K Mulligan, Kalli Kappel, et al. The rosetta all-atom energy function for macromolecular modeling and design. *Journal of chemical theory and computation*, 13(6):3031–3048, 2017.

Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic acids research*, 28(1): 235–242, 2000.

Jasper Butcher, Rohith Krishna, Raktim Mitra, Rafael I Brent, Yanjing Li, Nathaniel Corley, Paul T Kim, Jonathan Funk, Simon Mathis, Saman Salike, et al. De novo design of all-atom biomolecular interactions with rfdiffusion3. *bioRxiv*, 2025.

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.

Ruizhe Chen, Dongyu Xue, Xiangxin Zhou, Zaixiang Zheng, Xiangxiang Zeng, and Quanquan Gu. An all-atom generative model for designing protein complexes. *arXiv preprint arXiv:2504.13075*, 2025.

Chaoran Cheng, Jiaqi Guan, Milong Ren, Chengyue Gong, Cong Liu, Xinshi Chen, Ge Liu, and Wenzhi Xiao. A-code: Fully atomic protein co-design with unified multimodal diffusion. *arXiv preprint arXiv:2605.03360*, 2026.

Alexander E Chu, Jinho Kim, Lucy Cheng, Gina El Nesr, Minkai Xu, Richard W Shuai, and Po-Ssu Huang. An all-atom protein generative model. *Proceedings of the National Academy of Sciences*, 121(27):e2311500121, 2024.

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.

Justas Dauparas, Gyu Rie Lee, Robert Pecoraro, Linna An, Ivan Anishchenko, Cameron Glasscock, and David Baker. Atomic context-conditioned protein sequence design using ligandmpnn. *Nature Methods*, 22(4):717–723, 2025.

Kieran Didi, Zuobai Zhang, Guoqing Zhou, Danny Reidenbach, Zhonglin Cao, Sooyoung Cha, Tomas Geffner, Christian Dallago, Jian Tang, Michael Bronstein, et al. Scaling atomistic protein binder design with generative pretraining and test-time compute. In *International Conference on Learning Representations*, volume 2026, pp. 61056–61109, 2026.

Roland L Dunbrack Jr and Fred E Cohen. Bayesian statistical analysis of protein side-chain rotamer preferences. *Protein science*, 6(8):1661–1681, 1997.

Tomas Geffner, Kieran Didi, Zhonglin Cao, Danny Reidenbach, Zuobai Zhang, Christian Dallago, Emine Kucukbenli, Karsten Kreis, and Arash Vahdat. La-proteina: Atomistic protein generation via partially latent flow matching. In *International Conference on Learning Representations*, volume 2026, pp. 67350–67395, 2026.

Xiangzhe Kong, Yinjun Jia, Wenbing Huang, and Yang Liu. Full-atom peptide design with geometric latent diffusion. *Advances in Neural Information Processing Systems*, 37:74808–74839, 2024.

Xiangzhe Kong, Zishen Zhang, Ziting Zhang, Rui Jiao, Jianzhu Ma, Wenbing Huang, Kai Liu, and Yang Liu. Unimomo: Unified generative modeling of 3d molecules for de novo binder design. *arXiv preprint arXiv:2503.19300*, 2025.

Xiangzhe Kong, Junwei Chen, Ziting Zhang, Gaodeng Li, Qingyuan Zhu, Lei Wei, Mingyu Li, Yan Shi, Weiyang Dai, Zishen Zhang, et al. Programming biomolecular interactions with all-atom generative model. *bioRxiv*, pp. 2026–03, 2026.

Daniel Kovtun, Mehmet Akdel, Alexander Goncearenco, Guoqing Zhou, Graham Holt, David Baugher, Dejun Lin, Yusuf Adeshina, Thomas Castiglione, Xiaoyun Wang, et al. Pinder: the protein interaction dataset and evaluation resource. *bioRxiv*, pp. 2024–07, 2024.

Sidney Lyayuga Lisanza, Jacob Merle Gershon, Samuel WK Tipps, Jeremiah Nelson Sims, Lucas Arnoldt, Samuel J Hendel, Miriam K Simma, Ge Liu, Muna Yase, Hongwei Wu, et al. Multistate and functional protein design using rosettafold sequence space diffusion. *Nature biotechnology*, 43 (8):1288–1298, 2025.

Cong Liu, Milong Ren, Jiaqi Guan, Chengyue Gong, Jinyuan Sun, Xinshi Chen, and Wenzhi Xiao. Protdbench: A unified benchmark of protein binder design and evaluation. *arXiv preprint arXiv:2605.04118*, 2026.

Amy X Lu, Jason Yang, Yueming Long, Theophile Lambert, Wilson Yan, Sarah A Robinson, Kevin K Yang, Vladimir Gligorijevic, Kyunghyun Cho, Richard Bonneau, et al. Controllable all-atom protein generation with latent diffusion. *bioRxiv*, pp. 2024–12, 2024.

Tianyu Lu, Richard Shuai, Petr Kouba, Zhaoyang Li, Yilin Chen, Akio Shirali, Jinho Kim, and Po-Ssu Huang. Conditional protein structure generation with protpardelle-1c. *bioRxiv*, 2025.

Karolis Martinkus, Jan Ludwiczak, Wei-Ching Liang, Julien Lafrance-Vanasse, Isidro Hotzel, Arvind Rajpal, Yan Wu, Kyunghyun Cho, Richard Bonneau, Vladimir Gligorijevic, et al. Abdiffuser: full-atom generation of in-vitro functioning antibodies. *Advances in Neural Information Processing Systems*, 36:40729–40759, 2023.

Wei Qu, Jiawei Guan, Rui Ma, Ke Zhai, Weikun Wu, and Haobo Wang. P (all-atom) is unlocking new path for protein design. *bioRxiv*, pp. 2024–08, 2024.

Maxim V Shapovalov and Roland L Dunbrack. A smoothed backbone-dependent rotamer library for proteins derived from adaptive kernel density estimates and regressions. *Structure*, 19(6):844–858, 2011.

Richard W. Shuai, Talal Widatalla, Po-Ssu Huang, and Brian L. Hie. Sidechain conditioning and modeling for full-atom protein sequence design with fampnn. *bioRxiv*, 2025. doi: 10.1101/2025.02.13.637498. URL https://www.biorxiv.org/content/early/2025/02/17/2025.02.13.637498.

Hannes Stark, Felix Faltings, MinGyu Choi, Yuxin Xie, Eunsu Hur, Timothy O'Donnell, Anton Bushuiev, Talip Uçar, Saro Passaro, Weian Mao, et al. Boltzgen: Toward universal binder design. *BioRxiv*, 2025.

Protenix Team, Milong Ren, Jinyuan Sun, Jiaqi Guan, Cong Liu, Chengyue Gong, Yuzhe Wang, Lan Wang, Qixu Cai, Wenzhi Ma, et al. Pxdesign: Fast, modular, and accurate de novo design of protein binders. *BioRxiv*, pp. 2025–08, 2025.

Brian L Trippe, Jason Yim, Doug Tischer, David Baker, Tamara Broderick, Regina Barzilay, and Tommi Jaakkola. Diffusion probabilistic modeling of protein backbones in 3d for the motif-scaffolding problem. *arXiv preprint arXiv:2206.04119*, 2022.

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.

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.

Jingkai Zhang and Yuanyan Xiong. Packppi: An integrated framework for protein–protein complex side-chain packing and $\delta\delta$ g prediction based on diffusion model. *Protein Science*, 34(5):e70110, 2025.

Odin Zhang, Xujun Zhang, Haitao Lin, Cheng Tan, Qinghan Wang, Yuanle Mo, Qiantai Feng, Gang Du, Yuntao Yu, Zichang Jin, et al. Odesign: A world model for biomolecular interaction design. *arXiv preprint arXiv:2510.22304*, 2025.

A Related Work

Backbone generation and staged protein design. Diffusion and flow-based protein generators learn distributions over backbone geometry and support unconditional generation, motif scaffolding, and target-conditioned binder design (Trippe et al., 2022; Watson et al., 2023; Team et al., 2025). RFdiffusion demonstrated structure and function design through denoising in residue frames (Watson et al., 2023), while PXDesign provides a modular Protenix-based backbone generator for efficient binder design. These models establish increasingly strong global scaffolds. In many staged workflows, however, the generated backbone is fixed before inverse folding and side-chain packing. Consequently, backbone information passes to downstream sequence and atomistic models, while constraints discovered during full-atom design are not incorporated into the backbone-generation trajectory.

Sequence design and side-chain realization. Inverse folding methods like ProteinMPNN (Dauparas et al., 2022) assign amino-acid sequences to a given structure, but do not explicitly generate the side-chain conformations that realize the inferred interactions. Complementary side-chain packing methods search over backbone-dependent rotamer libraries using all-atom energy functions (Dunbrack Jr & Cohen, 1997; Shapovalov & Dunbrack, 2011; Alford et al., 2017). FAMPNN combines residue identity and conformation by factorizing each residue prediction into a categorical sequence decision and a conditional side-chain distribution, trained jointly with sequence and coordinate objectives (Shuai et al., 2025). Side-chain-aware design and packing improve the atomic realization of a supplied backbone, but retain the backbone as a fixed condition.

Joint all-atom generation. Recent algorithms increasingly model sequence and atomistic structure within a single generative system. Geometric latent diffusion has been applied to full-atom peptide design (Kong et al., 2024); AbDiffuser generates antibody sequence and full-atom structure (Martinkus et al., 2023); and Protpardelle jointly generates protein sequence and all-atom structure using residue-dependent side-chain states (Chu et al., 2024). RFdiffusion3, BoltzGen, and A-CODE extend atomistic generation to biomolecular interactions and protein co-design (Butcher et al., 2025; Stark et al., 2025; Cheng et al., 2026). Such models internalize the coupling among backbone, residue identity, and side-chain variables within a shared parameterization. However, existing approaches fail to elegantly resolve the coupling between residue identity and atomic structure. ProteoAA makes this dependency explicit through refinement, allowing the resulting side-chain geometry to feed back into backbone generation.

B Implementation Details

B.1 Representations and adapter interfaces

Coordinates, masks, and frames. Coordinates use an atom37 storage layout, in which each residue has thirty-seven named atom slots and unused slots are marked invalid. The atom-validity mask $\mathcal{A}$ records which slots hold a real atom and is recomputed after packing, so that atoms the side-chain module has just generated are treated as present. Residue-local frames are constructed from the (N, C_α, C) atoms of each residue. We write $\hat{F}_i$ for the frame of residue i built from the predicted backbone, and a coordinate expressed in $\hat{F}_i$ is the coordinate after mapping into that frame, which makes it invariant to the global rotation and translation of the structure. Residues whose three backbone atoms do not define a frame are flagged and receive no residual.

B2C Adapter. The B2C Adapter injects its residual into the side-chain module encoder output h_V, after full-atom encoding and before the sequence head and the side-chain denoiser, both of which read the fused representation. Injecting at this point rather than inside the side-chain denoiser is what makes the sequence term differentiable with respect to the adapter within the same pass. Because a full decode calls the encoder many times, each call derives the fused representation from the original h_V, so the residual does not accumulate across decoding steps. The adapter normalizes its input, concatenates a sinusoidal embedding of $\log \sigma_t$ of width 64, and applies a two-layer projection with hidden width 256 and a SiLU nonlinearity, whose output layer is zero-initialized. Projection widths must match the injection site, and the backbone source representation and the residual both have width 768.

Features for early conditioning. For each residue, $E_{\boldsymbol{\xi}^e}$ encodes every valid atom from an atom-slot embedding, which is a learned vector indexed by the atom37 slot the atom occupies and therefore identifies the atom by its chemical position rather than by its order in a list, together with the atom coordinate expressed in $\hat{F}_i$ and the predicted error of that atom with an indicator for whether an error was reported. These per-atom encodings are pooled by a masked mean and a masked maximum over the atoms the residue actually has. The pooled result is combined with a learned residue-identity embedding over the twenty canonical types and an unknown class, the realized torsions with an indicator for whether each torsion is defined, and a per-residue reliability block holding the mean and maximum predicted side-chain error together with frame-validity and coverage flags, giving the residue feature $\mathbf{U}_i$. The residue-identity embedding is parameterized separately from the one the late branch uses.

For each retained pair, the directed feature $\mathbf{V}_{ij}$ combines a radial-basis expansion of the interatomic distance, the offset direction rotated into $\hat{F}_i$, both atoms' predicted errors, and a backbone or side-chain indicator, pooled in the same way and concatenated with $\mathbf{U}_i$ and $\mathbf{U}_j$. Because the direction is expressed in the frame of the source residue rather than in a shared reference, the feature is not symmetric and $\mathbf{V}_{ij} \neq \mathbf{V}_{ji}$. Pairs are selected by a nearest-neighbour graph on the predicted C_α positions, and the selection is treated as non-differentiable preprocessing. The pair term is necessarily neighbourhood based, since a residue's own side chain expressed in its own frame adds nothing beyond its torsions, so any further signal must be cross-residue. The two output heads project to widths 384 and 128, matching $\mathbf{s}$ and $\mathbf{z}$, neither of which equals the token width of 768.

Features for late adaptation. $E_{\boldsymbol{\xi}^l}$ reads the residue representations obtained by re-encoding the generated full-atom structure with the full-atom MPNN, together with the realized torsions, compact contact and burial summaries involving the generated side chains, a residue-identity embedding, and the same reliability block. This re-encoding is performed with the B2C Adapter disabled.

Injection points. Within the diffusion module, $\mathbf{s}'$ is added to the token features before the diffusion transformer and is also supplied to each of its blocks. The pair residual $\mathbf{z}'$ is used as a pair bias in the transformer's global token attention and, after projection and broadcast into the sequence-local attention window, in the atom-attention encoder. The late residual is added to the normalized token representation that forms the input of the atom-attention decoder, which is the last point at which the token representation is read. Both routes read the same packed state and are injected at their respective sites during one corrective pass.

Masking and initialization. Within the early network, residuals are additionally zeroed at residues whose backbone frame is undefined and at neighbour pairs the graph did not retain. The masks $\mathbf{M}_B$ and $\mathbf{M}_{BB}$ of equation 3 are applied afterwards. In both cases, masking follows the output projection and its bias, so that positions outside the designed region receive exactly zero. Zero initialization applies to the residual output projections and biases of both routes. Each residual is further scaled by a fixed function of σ_t that equals one across the training window and tapers smoothly outside it.

B.2 Loss definitions

B2C Adapter masking conventions. A subset of residue identities in the designed region is masked, and all side chains of that region are removed from the input to $S_{\boldsymbol{\phi}}$, since a residue's side-chain coordinates determine its identity almost exactly and retaining them would let the sequence head read the quantity it is asked to predict. Side chains outside the designed region remain available as structural context. We retain an asymmetry inherited from the original objective of $S_{\boldsymbol{\phi}}$, in that the encoder observes masked identities while the side-chain denoiser is conditioned on ground-truth identities.

The design objective. In $\mathcal{L}_{\text{design}}$, the term $\mathcal{L}_{\text{aa}}$ is masked sequence cross-entropy and $\mathcal{L}_{\text{sc}}$ is a diffusion-weighted side-chain coordinate loss in residue-local frames. The sequence term is normalized by the number of residues in the example, and positions whose identity is unknown are excluded. Following the original implementation of $S_{\boldsymbol{\phi}}$, $0.1/21$ is added to every class of the one-hot target, which is then renormalized. This is equivalent to standard label smoothing with a coefficient of $0.1/1.1 \approx 0.091$ rather than 0.1, since the added mass raises the total before normalization

instead of being taken from the true class. The side-chain term is reduced over supervised coordinate components within each example, multiplied by that example's diffusion weight, and then averaged over the batch. Supervision follows the atom-slot and local-frame conventions of S_ϕ, including its treatment of padded slots, and is not limited to sequence-masked residues. Its diffusion weights use the internal side-chain noise level τ, which is distinct from the backbone level σ_t, and several values of τ are evaluated against the same fused representation within one example.

We fix $\lambda_{\mathrm{sc}} = 1$ and calibrate λ_{aa} once before training, on a batch drawn from the training split, so that the two terms contribute equal gradient magnitude at the adapter,

$$\lambda_{\mathrm{aa}} = \mathrm{clip}\left(\lambda_{\mathrm{sc}} \frac{\|\nabla_{\vartheta} \mathcal{L}_{\mathrm{sc}}\|}{\|\nabla_{\vartheta} \mathcal{L}_{\mathrm{aa}}\|},\ 10^{-3},\ 10^{3}\right). \tag{6}$$

The norms are taken over the adapter parameters alone, since the base modules are frozen and their gradients are not informative about the balance the adapter actually experiences. The two terms are not commensurate at unit weight, with the sequence term two to three orders of magnitude larger, so the calibrated value falls near 10^{-3}. The calibration routine reports both the raw gradient-ratio estimate and its clipped suggestion. Estimates outside $[10^{-3}, 10^{3}]$ are explicitly flagged for review and not adopted, since it would indicate that the two objectives cannot be balanced by reweighting.

Noise levels used in training. The B2C Adapter is trained at the single level $\sigma_t = 0.429$. The C2B adapter is trained with σ_t sampled per example from the discrete noise levels of the 400-step backbone diffusion schedule within $[\sigma_{\min}, \sigma_{\max}] = [0.1, 2.0]$ Å. The gate in equation 3 equals one throughout this window and smoothly decreases to zero in log-noise space over a multiplicative margin of two, reaching zero at $\sigma_t \leq 0.05$ Å and $\sigma_t \geq 4.0$ Å.

B.3 Sampling details

The realized noise level is the level presented to the denoiser after sampler churn, which for the published churn settings is twice the scheduled level across most of the trajectory. The level $\sigma^\star$ that selects the feedback step of Section 2.5 is matched against this realized level, and the step chosen is the one whose realized level is nearest $\sigma^\star$ rather than the first step to fall below it. The realized value is recorded and is the level on which the B2C Adapter and the C2B Adapter are conditioned at that step.

The reported sampler uses 400 solver steps over a schedule of 401 noise levels that terminates at $\sigma_{\min} \approx 6.8 \times 10^{-3}$. At the selected feedback event, a provisional backbone evaluation is followed by a corrective evaluation at the identical noisy coordinates and realized noise level. The solver then advances using the corrected estimate. Activating both branches of the C2B Adapter within that evaluation does not change this count. Full-atom generation may involve multiple internal sequence-unmasking and side-chain denoising steps, and these do not correspond to additional backbone feedback steps.

The side-chain module is called twice over a trajectory, once at the feedback step and once on the fully denoised backbone. The first call supplies feedback only. Its sequence and side-chain coordinates are not passed as context to the final call, since they were produced for a backbone the solver has since moved away from, so the reported structure is not anchored to a design built on an intermediate backbone. During final sequence and side-chain generation, the side-chain module encodes the completed backbone and reuses this cached residual. The B2C Adapter is not re-evaluated or conditioned on the terminal backbone noise level. see Appendix B.4.

The design and packing random-number streams are isolated from backbone sampling. A single full-atom decode invokes the encoder on the order of a hundred times, and without isolation those calls and the packing draws would advance the backbone sampler's own stream, so that the coupled run would diverge from the baseline sampler outside the feedback step for reasons unrelated to feedback. Generated side-chain coordinates contribute to feedback but are not included in the evolving noisy backbone state.

B.4 Noise-level sweep for B2C Adapter

B2C Adapter receives $\log \sigma_t$, so a single evaluation point licenses a claim only at that point. We therefore sweep σ_t across a range of backbone noise level and report per-level as well as pooled

results, since an adapter that helps at high noise and hurts at low noise is an outcome that pooling would hide.

At each noise level, every held-out structure is packed twice, once with the B2C Adapter enabled and once with it disabled, using the same noise realization and random seed. Because the backbone and $\mathbf{h}_{\text{res}}$ are computed before residual injection, the two runs are identical up to that point, isolating the effect of the adapter.

Side-chain accuracy is evaluated relative to each residue's local backbone frame. Before scoring, predicted side chains are mapped from their respective residue frames onto a common backbone, isolating side-chain conformation from backbone displacement. Backbone deviation is reported separately at each noise level as the root-mean-square deviation after Kabsch alignment, allowing changes in side-chain packing and backbone geometry to be assessed independently.

Table 4: B2C Adapter against the frozen system across backbone noise levels, reported as frozen $\rightarrow$ adapted. Side-chain root-mean-square deviation (scRMSD), rotamer recovery, and side-chain to side-chain lDDT are computed relative to each residue's own backbone. Backbone deviation is that of the shared proposal and is identical in both arms. Entries given as Δ are those for which only the difference was recorded.

Set	σ_t	bb RMSD (Å)	scRMSD (Å)	Rotamer rec.	lDDT sc–sc	Bad bonds
AFDB	0.010	0.014	1.062 → 0.979	0.7884 → 0.8045	0.8783 → 0.9085	0.0007 → 0.0017
AFDB	0.082	0.068	1.147 → 1.067	0.7605 → 0.7836	0.8656 → 0.8976	0.0007 → 0.0018
AFDB	0.429	0.226	1.317 → 1.246	0.7168 → 0.7466	0.8362 → 0.8701	0.0010 → 0.0030
AFDB	1.642	0.608	1.556 → 1.500	0.6490 → 0.6923	0.7923 → 0.8263	0.0024 → 0.0047
AFDB	4.881	1.625	2.165 → 1.848	0.4675 → 0.6073	0.6897 → 0.7609	0.0598 → 0.0045
AFDB	pooled	—	1.502 → 1.365	0.6765 → 0.7268	0.8124 → 0.8527	0.0129 → 0.0031
CASP14+15	0.010	0.016	1.394 → 1.525	0.6506 → 0.5932	0.8534 → 0.8143	0.0006 → 0.0057
CASP14+15	0.082	0.109	1.496 → 1.651	$\Delta -0.0371$	0.8263 → 0.7950	0.0032 → 0.0053
CASP14+15	0.429	0.469	2.129 → 1.947	0.4425 → 0.4855	0.7372 → 0.7489	0.0728 → 0.0139
CASP14+15	1.642	1.502	2.472 → 2.198	$\Delta +0.0651$	0.6813 → 0.7068	0.0997 → 0.0315
CASP14+15	4.881	3.943	2.725 → 2.352	0.2581 → 0.3805	0.6132 → 0.6701	0.1377 → 0.0335
CASP14+15	pooled	—	2.110 → 1.974	$\Delta +0.0272$	0.7423 → 0.7470	0.0628 → 0.0180
CASP16	0.010	0.017	1.387 → 1.548	0.6593 → 0.5839	0.8568 → 0.8057	0.0009 → 0.0077
CASP16	0.082	0.106	1.510 → 1.670	$\Delta -0.0545$	0.8257 → 0.7850	0.0038 → 0.0079
CASP16	0.429	0.463	2.293 → 2.022	0.4151 → 0.4668	0.7037 → 0.7247	0.0964 → 0.0197
CASP16	1.642	1.504	2.644 → 2.314	$\Delta +0.0764$	0.6499 → 0.6833	0.1258 → 0.0419
CASP16	4.881	3.902	2.804 → 2.462	0.2601 → 0.3737	0.6037 → 0.6530	0.1353 → 0.0464
CASP16	pooled	—	2.205 → 2.072	$\Delta +0.0224$	0.7279 → 0.7303	0.0724 → 0.0247

Two patterns hold across the sweep. First, the benefit of adaptation grows with the noise level. On every set, the improvement in side-chain accuracy and in rotamer recovery is largest at $\sigma_t = 4.881$ and smallest at the bottom of the window, and the same ordering holds for the bad-bond fraction, which the adapter increases slightly at low noise and reduces by an order of magnitude at high noise. This is the behaviour the mechanism predicts, in that a backbone that is already almost determined leaves little for a conditioning signal to supply, whereas a heavily corrupted backbone is where information about the intended packing is most useful.

Second, the sign of the effect at low noise depends on the distribution. On the in-distribution set, the adapter improves side-chain accuracy at every level. On the CASP sets it degrades accuracy at $\sigma_t = 0.010$ and $\sigma_t = 0.082$ and improves it at and above $\sigma_t = 0.429$, so the two arms cross between those levels. We therefore train and apply the adapter at $\sigma_t = 0.429$, which is also the level the sampler reaches at the step where feedback is applied. The resulting residual is cached and reused during final decoding without re-evaluating the adapter at the terminal backbone noise level. We do not claim that adaptation is beneficial at arbitrarily low noise.

C Side chain packing results on whole-complex scoring

Evaluation Set Statistics. Our model was trained on Teddymer (Didi et al., 2026), a new large-scale dataset of synthetic binder-target pairs. We construct the evaluation set from complexes sampled from PDB (Berman et al., 2000) and PINDER-S (Kovtun et al., 2024), yielding a total of 281 complexes. We did additional filtering for quality and leakage reduction.

Table 5: **Binder-only side-chain packing evaluation.** We evaluate side-chain packing on the binder while keeping the receptor fixed as structural context using Proteo-AA corrected backbone. The native-sequence setting serves as a reference ceiling, in which each side-chain packing model is given the ground-truth binder backbone. ↓ indicates lower is better and ↑ indicates higher is better.

Metric	Native	FaMPNN	LigandMPNN	PackPPI	Proteo-AA
Sym. RMSD ↓	1.14922	1.45540	1.55973	1.60861	**1.43311**
χ Rec.@40 ↑	0.80647	**0.77111**	0.75390	0.73098	0.76270
χ Rec.@20 ↑	0.72102	0.65447	0.64895	0.61726	**0.66516**
Rot. Rec. ↑	0.72235	**0.67540**	0.64342	0.61844	0.66276
Rot. Outlier ↓	0.27765	**0.32460**	0.35658	0.38156	0.33724
χ_1@20 ↑	0.81914	0.74736	0.72421	0.69816	**0.75469**
χ_1@40 ↑	0.87670	0.82782	0.80582	0.78639	**0.83341**
$\chi_{1,2}$@20 ↑	0.66358	0.56174	0.54927	0.51279	**0.57692**
$\mathrm{lDDT}_{\mathrm{SC-SC}}$ ↑	0.88143	0.82282	0.79538	0.78560	**0.82943**
$\mathrm{lDDT}_{\mathrm{SC-Env}}$ ↑	0.90894	0.86244	0.84494	0.83237	**0.87235**
Bond MAE ↓	0.02974	0.05846	0.04911	**0.04900**	0.05829
Bad Bond Frac. ↓	0.00024	0.05397	0.05035	0.05296	**0.05022**
Complete. ↑	1.00000	**1.00000**	**1.00000**	0.99990	**1.00000**

Evaluation set filtering To minimize set leakage, we remove complexes that overlap with our training data based on both structure identifiers and structural similarity, excluding any complex with a corresponding training-chain match with TM-score ≥ 0.5. For baseline methods, we similarly minimize overlap using publicly available information about their training data. We further filter the remaining complexes for structural and interface quality using the following criteria: (1) experimental resolution ≤ 2.5 Å, (2) buried surface area ≥ 500 Å^2, and (3) at least 10 interface residues.

Protocol. Each complex is evaluated at five fixed noise levels $\sigma \in \{0.0100, 0.0815, 0.4293, 1.6422, 4.8814\}$ Å. Each σ is an independent Gaussian corruption of the deposited backbone, $x_\sigma = x_{\text{native}} + \sigma\,\epsilon$ with $\epsilon \sim \mathcal{N}(0, I)$, drawn once per (complex, σ) and shared by every method so that methods differ only in the model. The values are exact steps of PXDesign's published inference scheduler lying inside $[0.01, 5.0]$ Å, the final $\sim 30\%$ of its trajectory. Therefore, we ensure a range where produced backbones are informative to side chain packing.

In the *binder-only* setting, the receptor is held at its deposited coordinates, and only binder atoms are corrupted. The same Proteo-AA correct backbone is supplied to all methods for side-chain packing.

Only binder residues are scored, but the receptor is retained as partner atoms in the environment term, so $\mathrm{lDDT}_{\text{SC-Env}}$ still measures whether the binder's side chains fit the surface they were built against. The *whole-complex* variant in Table 7 instead corrupts every chain and scores all residues, and is reported for completeness.

Metrics. **Sym. RMSD** is a symmetry-aware side-chain RMSD, resolving the indistinguishable-atom ambiguity in seven residue types (ARG, ASP, GLU, LEU, VAL, PHE, TYR). χ **Rec.@**k is the fraction of χ angles within k° of native, and χ_1**@**k, $\chi_{1,2}$**@**k the same restricted to the first, and first two, angles. **Rot. Rec.** is the fraction of residues whose *every* χ angle falls within 40°; **Rot. Outlier** is its complement, $1 -$ Rot. Rec. — despite the name, it is native-referenced and is not a rotamer-library occupancy test. **lDDT**$_{\text{SC-SC}}$ scores binder side-chain atoms against binder side-chain atoms and **lDDT**$_{\text{SC-Env}}$ against the whole complex. **Bond MAE** and **Bad Bond Frac.** are reference-free covalent geometry checks, and **Complete.** the fraction of expected side-chain atoms that were produced. All are computed with the project's canonical implementation, unmodified.

D Inference time comparison

In this section, Table 8 reports inference time across various methods inherited from the ACODE table. All methods are run on an NVIDIA A100-80G GPU.

Table 6: **Binder side-chain packing on a shared PXDesign backbone.** Every method receives the *same* PXDesign originated backbone; the receptor keeps its deposited side chains as fixed context and only the binder's side chains are predicted and scored. ↓ lower is better, ↑ higher is better.

Metric	Native	FaMPNN	LigandMPNN	PackPPI	Proteo-AA
Sym. RMSD ↓	1.14922	2.1959	2.2813	2.3033	**1.52066**
χ Rec.@40 ↑	0.80647	**0.7333**	0.7287	0.7065	0.73048
χ Rec.@20 ↑	0.72102	0.6207	0.6190	0.5889	**0.63549**
Rot. Rec. ↑	0.72235	**0.6357**	0.6196	0.5936	0.62462
Rot. Outlier ↓	0.27765	**0.3643**	0.3804	0.4064	0.37538
χ_1@20 ↑	0.81914	0.7075	0.6916	0.6647	**0.71794**
χ_1@40 ↑	0.87670	**0.7933**	0.7794	0.7602	0.78475
$\chi_{1,2}$@20 ↑	0.66358	0.5172	0.5113	0.4780	**0.54766**
lDDT$_{\mathrm{SC-SC}}$ ↑	0.88143	0.7967	0.7720	0.7607	**0.82094**
lDDT$_{\mathrm{SC-Env}}$ ↑	0.90894	0.8339	0.8143	0.8024	**0.86114**
Bond MAE ↓	0.02974	0.0988	0.0875	0.0874	**0.03013**
Bad Bond Frac. ↓	0.00024	0.0757	0.0735	0.0734	**0.00090**
Complete. ↑	1.00000	**1.0000**	**1.0000**	0.9999	**1.00000**

Table 7: **Whole-complex side-chain packing evaluation.** We pack and evaluate side chains over the entire protein complex. The native column provides a reference ceiling. ↓ indicates lower is better and ↑ indicates higher is better.

Metric	Native	FaMPNN	LigandMPNN	Proteo-AA
Sym. RMSD ↓	1.18331	1.5983	1.7513	**1.38987**
χ Rec.@40 ↑	0.80779	0.7187	0.7067	**0.75874**
χ Rec.@20 ↑	0.72279	0.6239	0.6098	**0.67109**
Rot. Rec. ↑	0.72455	0.6136	0.5827	**0.66479**
Rot. Outlier ↓	0.27545	0.3864	0.4173	**0.33521**
χ_1@20 ↑	0.82106	0.7006	0.6717	**0.75843**
χ_1@40 ↑	0.87784	0.7677	0.7431	**0.81508**
$\chi_{1,2}$@20 ↑	0.66679	0.5300	0.4959	**0.59663**
lDDT$_{\mathrm{SC-SC}}$ ↑	0.88001	0.8083	0.7739	**0.84324**
lDDT$_{\mathrm{SC-Env}}$ ↑	0.90939	0.8517	0.8238	**0.87984**
Bond MAE ↓	0.02974	0.0312	**0.0159**	0.03051
Bad Bond Frac. ↓	0.00021	0.0025	0.0015	**0.00124**
Complete. ↑	1.00000	1.0000	1.0000	1.0000

Table 8: Inference time (seconds) per sample across different protein lengths on a single NVIDIA A100-80G GPU. Lower is better.

Model	Sequence Length				
	100	200	300	400	500
A-CODE	2.9	5.2	7.8	10.9	14.6
La-Proteina	1.7	4.1	7.5	12.0	17.6
La-Proteina (w/ tri)	7.7	28.5	65.7	168.8	182.0
MultiFlow	23.5	25.8	30.5	44.9	61.0
Proteo-AA	12.3	12.7	15.0	17.1	23.7