PLDESIGN: UNIFIES ALL-ATOM COMPLEX REPRESENTATION AND GEOMETRY-GUIDED PROTEIN DESIGN

Anonymous authors
Paper under double-blind review

ABSTRACT

Modeling protein complexes requires coordinating atomic detail with cross-chain geometry, placing substantial demands on generative models of interacting systems. Autoencoder-based structural compression offers a way to separate atomic reconstruction from distribution learning, opening the possibility of modeling these coupled relationships in a compact space. We introduce **PLDesign** to investigate this direction through joint encoding of complete all-atom complexes. An Atom14 autoencoder learns a shared representation that preserves intrachain structure and interchain organization, while latent perturbations reveal a correspondence with decoded geometric variation. A downstream latent flow model uses this representation for unconditional backbone generation and target-conditioned binder design. Geometry-guided trajectory search further makes the latent information actionable during sampling, ranking intermediate states before coordinate decoding or external structure prediction. This search improves unconditional designability and enriches candidates in a matched computational binder screen, whose retained complexes contain locally packed interfaces with potential value for interface seed development. PLDesign connects joint complex encoding with latent-guided exploration, opening a route to studying and designing interacting molecular systems through compressed structural representations.

1 INTRODUCTION

Protein interactions depend on the local atomic configurations of the participating chains and their relative arrangement. Engineering this spatial organization is central to binder design, which seeks selective target recognition for therapeutic development and biological research (Huang et al., 2016; Kuhlman & Bradley, 2019; Bennett et al., 2023). A useful structural representation must therefore preserve these coupled geometric relationships, making them accessible for analysis and the exploration of new candidates.

Coordinate-space methods, including BoltzGen, PXDesign, and RFdiffusion3, expose distances and steric contacts through explicit three-dimensional positions (Stark et al., 2025; Team et al., 2025; Butcher et al., 2025). As complexes grow, modeling atomic detail and cross-chain relationships requires substantial computation while placing multiple geometric constraints on the generator. Learned structural compression offers an alternative, using an autoencoder to capture geometric information in a compact representation and a downstream generative model to learn its distribution (Li et al., 2025; Geffner et al., 2026). By moving distribution learning and sampling into this latent space, the formulation can reduce computational costs while leaving atomic reconstruction to the decoder. The potential computational savings motivate extending autoencoder-based structural compression to larger, interacting systems. For protein complexes, the central question is whether a compact representation can jointly capture intrachain structure and the cross-chain geometry that defines their interfaces.

Recent latent approaches illustrate how learned structural compression can support binder design. Proteina-Complexa combines autoencoder-based atomistic generation with inference-time optimization (Didi et al., 2026), while UniMoMo supports binder design across molecular types within a shared latent framework (Kong et al., 2025). At the encoding stage, Complexa retains a monomer

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.

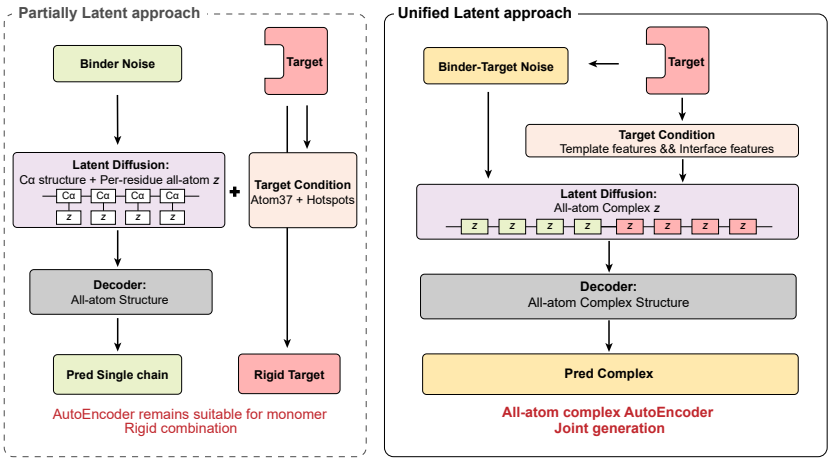

Figure 1: Structural encoding strategies for binder design. PLDesign learns residue-level codes from complete Atom14 complexes, placing both partners in a shared representation. A downstream flow model and geometric trajectory search use these codes for generation and inference-time candidate selection.

autoencoder, while UniMoMo represents the binder and binding site independently. This leaves open whether encoding complete complexes together can preserve intrachain structure and cross-chain geometry in a unified latent representation.

PLDesign investigates unified complex modeling through an Atom14 autoencoder, a downstream latent flow model, and geometry-guided trajectory search (Figure 1). Reconstruction and cross-chain isolation analyses characterize the structural information retained by joint encoding, while controlled perturbations connect latent variation with decoded geometry. Trajectory search tests the utility of these geometric signals, improving unconditional designability and enriching candidates in a matched target-conditioned experiment. Runtime and memory measurements assess the computational cost of generation with search, and interface analyses examine potential design uses of the retained complexes. Together, the experiments support the feasibility of joint complex encoding and the use of latent geometry for candidate selection, providing an empirical basis for further exploration of interacting systems through compressed structural representations.

Key contributions

- **Unified representation of all-atom complexes.** We demonstrate that a continuous autoencoder can retain intrachain geometry and cross-chain organization within compact residue-level codes, using reconstruction and context-isolation analyses to characterize the learned information.
- **Latent geometry for inference-time search.** Perturbation analyses motivate geometric trajectory scores, whose utility is tested through unconditional designability and conditional candidate enrichment. Runtime and memory measurements characterize the computational cost of generation with latent search.
- **Evidence for generative complex modeling.** We evaluate full-pipeline yield, local interfaces, scaffold diversity, and structural reuse to characterize the current latent generator. These observations identify opportunities for interface seed development and directions for organizing complex representations for generation.

2 Related Work

Flow matching for protein generation. Flow matching trains a continuous-time velocity field along a chosen probability path and provides a simulation-free route to generative modeling (Lipman et al., 2022; Liu et al., 2022). For proteins, the field has progressed from backbone and frame-space models such as FrameFlow and FoldFlow (Yim et al., 2023; Bose et al., 2024) to larger-scale

backbone generators such as Proteina (Geffner et al., 2025), and then to all-atom or latent methods. ProteinZen combines latent and SE(3) flow matching for all-atom generation (Li & Kortemme, 2024). La-Proteina keeps C_α coordinates explicit while encoding sequence and side-chain detail into per-residue latents (Geffner et al., 2026). ProteinAE maps backbone atoms into a fully continuous latent with a single flow-matching loss (Li et al., 2025). These models establish continuous structural latents as a generative substrate. PLDesign examines how complex-level encoding can additionally support interaction-aware representation and geometric search.

Protein structure representations. Discrete tokenizers such as the ESM3 structural VQ-VAE, DPLM-2, Foldseek 3Di, ProToken, and GCP-VQVAE connect compact structural codes with sequence models (Hayes et al., 2025; Wang et al., 2025; Van Kempen et al., 2024; Lin et al., 2023b; Pourmirzaei et al., 2025). Continuous and partially latent autoencoders such as ProteinMAE, La-Proteina, Kanzi, and ProteinAE retain fine geometric detail for generation and evaluation (Yuan et al., 2023; Geffner et al., 2026; Dilip et al., 2026; Li et al., 2025). Chain-aware FoldToken-4 models reconstruct multichain backbones with discrete structural codes (Gao et al., 2024). A-CODE uses an Atom14 layout for joint atom-type and coordinate generation (Cheng et al., 2026). PLDesign combines a continuous per-residue latent, Atom14 complex reconstruction, explicit interface features, and joint multichain training.

Conditional binder generation. Coordinate-space systems including RFdiffusion, Chroma, BoltzGen, PXDesign, and RFdiffusion3 generate target-conditioned structures with explicit geometric control (Watson et al., 2023; Ingraham et al., 2023; Stark et al., 2025; Team et al., 2025; Butcher et al., 2025). Proteina-Complexa combines a monomer autoencoder with conditional flow generation and test-time compute (Didi et al., 2026). UniMoMo performs conditional diffusion over separately encoded molecular blocks (Kong et al., 2025). PLDesign learns structural codes from complete bound complexes and evaluates latent-only trajectory scoring alongside the yield and interface properties of conditional generation.

3 METHODS

3.1 UNIFIED COMPLEX REPRESENTATION

PLDesign separates structural encoding, latent generation, and inference-time trajectory selection. A continuous autoencoder (E_ψ, D_ϕ) first learns a shared representation of interacting protein chains. For an Atom14 complex $x = (x_T, x_B) \in \mathbb{R}^{L\times 14\times 3}$ with $L = L_T + L_B$, the encoder produces a per-residue sequence $z = E_\psi(x) \in \mathbb{R}^{L\times 16}$. Both partners are processed together, so each residue code can depend on its chain structure and the surrounding interaction context. A coordinate-space flow decoder reconstructs atomic positions under a masked reconstruction objective with stereochemical auxiliary losses. The autoencoder and downstream latent model use AF3-style Diffusion Transformers (Figure 2). Appendices B and E specify the Atom14 layout, masking rules, and reconstruction loss.

3.2 LEARNING INTERACTION-AWARE STRUCTURAL CODES

Interface Conditioning Features (ICF) organize interaction context within the pair representation. The module combines chain identities, hotspot annotations, and reference geometry into a layered map whose projected features bias attention toward interface neighborhoods. A zero-initialized gate controls the map's contribution, while joint encoding lets information pass between interacting chains. Atom14 supervision trains the code to retain both backbone geometry and side-chain detail. The ICF comparison examines how this feature extraction affects reconstruction under a fixed complex-AE architecture. Appendix C provides the module inputs and projection details.

3.3 LATENT GEOMETRY AND CANDIDATE SCORING

The relationship between latent displacement and decoded geometry motivates scoring intermediate states directly in the learned representation. Our scores measure local regularity of consecutive binder codes, directional consistency, and agreement with the current candidate ensemble. Branching explores alternative paths, pruning removes low-scoring groups, and backtracking allocates ad-

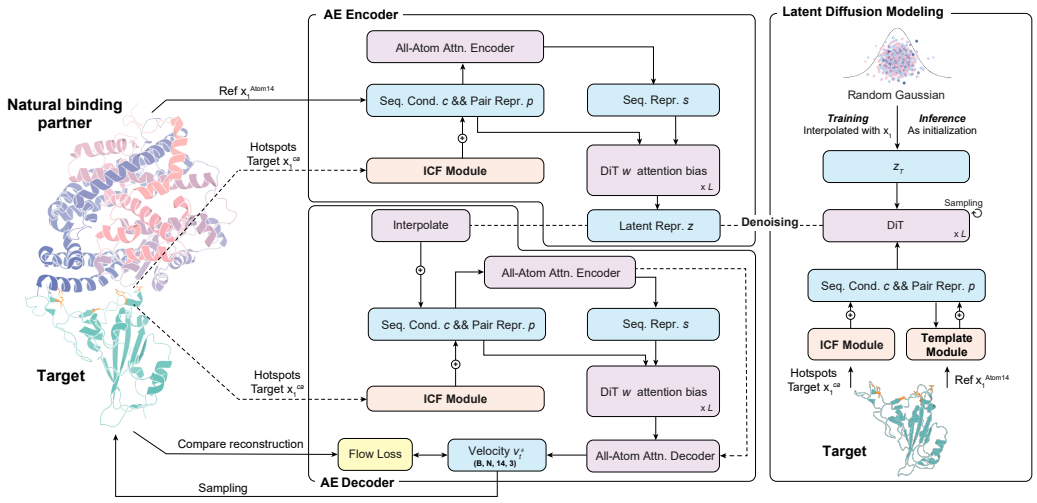

Figure 2: PLDesign architecture. The complex autoencoder learns per-residue structural codes with cross-chain context supplied through joint attention and interface features. A downstream latent flow model consumes these codes under target-template and hotspot conditioning.

ditional samples around selected states. The model weights remain fixed while these operations redistribute sampling effort at inference time. Because branch ranking uses latent features, it requires neither coordinate decoding nor external structure prediction. Appendix K defines the scores and search schedule.

For binder design, selected trajectories are decoded into backbone candidates for ProteinMPNN sequence assignment and AF3 complex prediction. Within these candidates, local binder-side patches and their placement against the target provide potential starting points for interface development. We call such a patch together with its relative placement an interface seed. Search selects complete candidates, whereas seed-level analysis concerns the local interaction geometry available for subsequent design.

3.4 Downstream Latent Generation

With the autoencoder frozen, a protein latent diffusion model (PLDM) learns to sample structural codes for unconditional backbones and target-conditioned complexes. We describe the conditional formulation below. Let $c_{\text{cond}} = (h, x_T^{\text{tmpl}})$ collect the hotspot field and target template. The conditional model factorizes as

$$p_\theta(x_T, x_B \mid c_{\text{cond}}) = \int D_\phi(x_T, x_B \mid z)\, p_\varphi(z \mid c_{\text{cond}})\, \mathrm{d}z. \tag{1}$$

The latent flow updates target and binder tokens together, while a soft template condition supplies the reference target geometry. Both partners therefore belong to the sampled state, while the conditioning pathway supplies hotspots and the reference target geometry.

For a training structure, we obtain $z_1 = E_\psi(x_1)$ and draw $z_0 \sim \mathcal{N}(0, I)$. Rectified flow matching uses $z_t = (1-t)z_0 + tz_1$, with a shared time for both chains (Lipman et al., 2022; Liu et al., 2022). The vector field v_φ gives the endpoint estimate

$$\hat{z}_1 = z_t + (1-t)v_\varphi(z_t, t, c_{\text{cond}}). \tag{2}$$

Training averages endpoint errors separately over the two chains and adds decoded-coordinate supervision. Appendix F specifies loss weighting and the training schedule. Stochastic integration supplies the candidate trajectories on which latent search operates, and the frozen decoder converts selected outputs into coordinates. The sampling equation and integration settings appear in Appendix G.

4 EXPERIMENTS

We evaluate whether joint compression preserves complex geometry, whether latent features support inference-time selection, and what the resulting generator produces. Reconstruction and perturbation analyses establish the representational evidence, followed by generation, interface, and computational-cost evaluations.

4.1 DATASET AND MODEL CONFIGURATION

PLDesign was trained on PDB biological dimers, PDB monomers, and Foldseek-clustered AFDB structures with sampling weights of 40%, 10%, and 50%. The PDB partition contains 62,086 records. The AFDB component was obtained from the clustered and filtered dataset released with La-Proteina (Geffner et al., 2026). Appendices A and F describe data processing, filtering, crop lengths, training budgets, and representative coverage. Table 7 provides the architectural hyperparameters.

4.2 STRUCTURAL FIDELITY OF THE COMPLEX AUTOENCODER

We evaluate complex reconstruction on the CASP14/CASP15 datasets with PLDesign, ProteinAE, and the multichain-aware FoldToken-4 (Li et al., 2025; Gao et al., 2024). The CASP structures were excluded from gradient-based optimization, and the model configurations were determined independently of their reconstruction metrics.

Table 1: CASP14/15 dimer reconstruction ($n = 39/153$, totaling 192 physical pairs). Backbone DockQ uses the four-method valid intersection ($n = 25/138$). PLDesign (split) disables cross-chain pathways and centers the chains separately. ProteinAE/FoldToken-4 provide no all-atom results ($-$). Every model receives the same filtered complex list and contributes one reconstruction per input under the same chain mapping, alignment, and metric implementation. Entries report mean $\pm$ sample standard deviation (three decimals). Arrows indicate preference. Bold and underline mark the best/second-best eligible results per benchmark and metric.

	RMSD (Å) ↓			lDDT ↑			$\mathrm{DockQ_{BB}}$ ↑
Method	Cα	Backbone	All-atom	Cα	Backbone	All-atom	Intersection
CASP14							
PLDesign	**0.505±0.104**	**0.659±0.113**	**0.754±0.086**	**0.986±0.006**	**0.969±0.013**	**0.919±0.020**	**0.865±0.078**
PLDesign (split)	1.564±0.084	1.659±0.091	1.793±0.153	0.701±0.008	0.686±0.007	0.667±0.014	0.693±0.084
ProteinAE	0.963±0.514	1.247±0.529	–	0.960±0.019	0.959±0.019	–	0.821±0.135
FoldToken-4	4.314±4.881	4.282±4.885	–	0.757±0.142	0.769±0.138	–	0.501±0.246
CASP15							
PLDesign	**0.489±0.094**	**0.618±0.142**	**0.708±0.102**	**0.990±0.007**	**0.979±0.009**	**0.932±0.008**	**0.916±0.060**
PLDesign (split)	1.573±0.045	1.650±0.086	1.725±0.088	0.702±0.007	0.688±0.005	0.675±0.006	0.684±0.080
ProteinAE	0.818±0.317	1.122±0.370	–	0.972±0.016	0.971±0.016	–	0.900±0.088
FoldToken-4	2.634±1.646	2.598±1.649	–	0.789±0.063	0.801±0.060	–	0.552±0.160

Reconstruction quality. We evaluate atomic reconstruction and interchain organization on CASP14 and CASP15 dimers (Table 1). Compared with ProteinAE, PLDesign reduces backbone RMSD by approximately 45–47% and improves backbone DockQ on the common valid intersections, indicating better recovery of both chain geometry and relative placement. It also outperforms the discrete-code FoldToken-4 baseline on the shared reconstruction metrics. At all-atom resolution, PLDesign achieves RMSDs of 0.754 and 0.708 Å on CASP14 and CASP15, respectively, supporting the feasibility of retaining atomic detail and interchain organization within the unified code.

To assess latent capacity, we compare autoencoders with 8, 16, and 32 channels per residue. D32 gives the lowest dimer C_α and backbone errors, while D16 achieves the best all-atom reconstruction on both CASP benchmarks (Table 11). We use D16 to provide the downstream LDM with accurate all-atom structural information, including the side-chain context relevant to interfaces.

On the CASP14/15 monomer subsets, PLDesign achieves backbone reconstruction accuracy broadly comparable to that of ProteinAE, while Atom14 supervision extends reconstruction to side-chain

coordinates (Table 10). Appendix I further examines latent width, multichain training, training duration, atom-level supervision, and ICF feature extraction.

Cross-chain context in complex reconstruction. PLDesign (split) probes dependence on interaction context by disabling cross-chain attention and pair conditioning in both the encoder and decoder. ICF and contact inputs are also removed, and the two chains are centered separately, so each is processed without its partner's context. Under this intervention, all-atom RMSD increases approximately 2.4-fold on both CASP benchmarks, while backbone DockQ and lDDT at every evaluated atomic resolution decline (Table 1). These changes show the model's dependence on complex context, with the observed effect reflecting both cross-chain information removal and separate coordinate centering.

Within the joint complex model, we further assess explicit interface features while keeping the DiT architecture, training data, optimization budget, and evaluation pipeline fixed. Simple ICF yields lower all-atom RMSD and higher DockQ on the dimer benchmarks, whereas No ICF achieves lower C_α RMSD (Table 14). The attention maps show interchain bands even without ICF, with Simple ICF producing more concentrated bands and clearer intrachain patterns (Figure 5). By supplying interface-neighborhood information without specifying side-chain conformations, ICF may ease the inference of cross-chain relationships and indirectly benefit atomic reconstruction. Across all five monomer subsets, Simple ICF also lowers all-atom RMSD but raises C_α RMSD, while changes in backbone RMSD are mixed (Table 15).

Structured and smooth latent geometry. We next examine how structural relationships are organized within the code, because accessible latent geometry would provide a basis for candidate selection. The t-SNE visualization in Figure 3B groups structurally similar complexes in neighboring regions. This qualitative pattern motivates a more direct test of whether changes in latent codes correspond to changes in decoded structures. We therefore apply controlled perturbations and relate latent displacement to DockQ, interface pair-distance error, binder radius of gyration, and smooth lDDT across perturbation levels (Figure 3C). The observed correlations suggest that latent variation carries information about both interface organization and broader structural geometry. These associations motivate relative latent scores for intermediate-state ranking. Trajectory-search experiments then test whether the geometric signals can guide sampling toward more useful outputs before external evaluation.

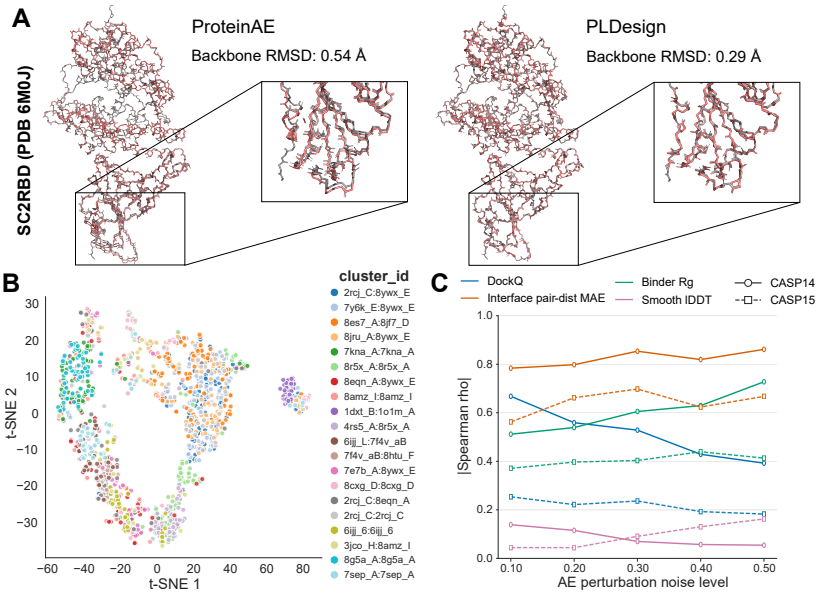

Figure 3: PLDesign autoencoder analysis. Panel A compares reconstruction with ProteinAE. Panel B shows a t-SNE projection of training-set latents grouped by chain similarity. Panel C relates latent perturbation magnitude to interface geometry across noise levels.

4.3 INFERENCE-TIME SEARCH AND DOWNSTREAM GENERATION

Latent search tests whether the learned geometry can improve sample selection without changing model parameters. The unconditional benchmark first isolates this use of structural information from cross-chain conditioning. Target-conditioned experiments then characterize candidate enrichment, complete-pipeline yield, and local interface properties, while efficiency measurements quantify the cost of search (Figure 4).

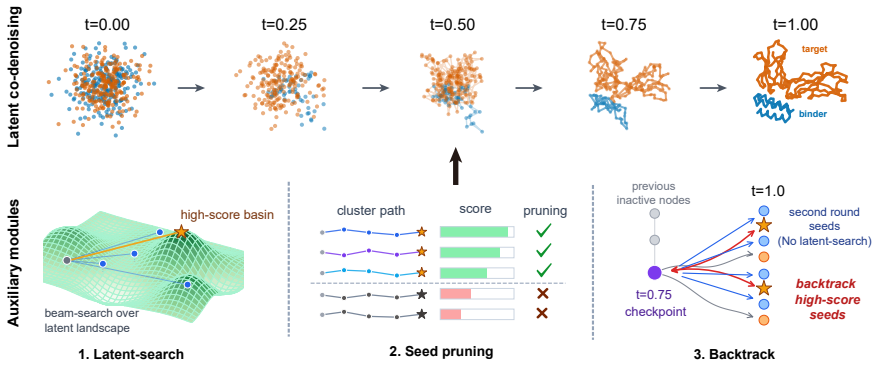

Figure 4: Latent geometry guides trajectory search in PLDesign. The upper panel depicts a downstream sampler that updates target and binder codes under reference conditioning. The lower panel shows how latent scores guide branching, pruning, and backtracking before coordinate decoding.

Unconditional backbone generation. We test the utility of latent geometry for candidate selection on 690 backbones, with ten samples at each integer length from 60 to 128 residues. Direct sampling and search use the same PLDM checkpoint and ProteinMPNN–ESMFold evaluation protocol (Table 2). Search increases the number of designable structures from 367 (53%) to 510 (74%) under an identical final output budget. DPT rises from 0.16 to 0.29, indicating that improved designability accompanies reduced diversity within the passing subset. The search result exceeds the reported designability of ESM3, DPLM-2, and LSD, while direct sampling has the lowest reported DPT. The within-model comparison demonstrates an inference-time benefit from latent-guided selection under a fixed final output budget, with search costs characterized separately in the PD-L1 setting below.

Trajectory ranking in PLDesign uses no external structure-prediction feedback. ProteinMPNN and ESMFold are applied only after selection, allowing the experiment to assess the utility of latent geometry for candidate prioritization. Proteina-Complexa incorporates structure-prediction scores into its search (Didi et al., 2026), introducing an additional source of guidance related to downstream structural evaluation. We therefore report its no-search result in Table 2 as a reference for generation without predictor-guided selection.

Target-conditional evaluation protocol. We evaluate PLDesign, BoltzGen, PXDesign, and RFDiffusion3 on six protein targets used in prior computational binder-design studies (Stark et al., 2025; Team et al., 2025; Butcher et al., 2025). Each method contributes 100 planned candidates per cell on the shared target–length grid, totaling 8,000 candidates across 80 cells. Generated backbones undergo ProteinMPNN binder sequence design and AF3 complex prediction using the reference target as a template. The common screen retains 252 PLDesign candidates (3.15%), compared with 2,583 for BoltzGen (32.29%), 4,876 for PXDesign (60.95%), and 3,341 for RFDiffusion3 (41.76%). A separate matched PD-L1 comparison shows search-based enrichment from 85/1,500 (5.67%) to 114/1,500 (7.60%) screen-passing candidates, with gains varying by binder length. The retained six-target cohorts provide the basis for subsequent analysis of local interface geometry. Appendix L details the screening criteria, raw-coordinate analyses, and standard AF3 confidence metrics.

Local interfaces within retained candidates. To examine local interaction geometry, we analyze the AF3-predicted interfaces of candidates that pass the shared structural screen (Bennett et al.,

Table 2: Unconditional generation with designability (Des ↑) and designable pairwise TM (DPT ↓). PLDesign direct/search share a checkpoint and 690-output budget. Search DPT weights all 69 lengths equally. Baselines follow prior reports (Li et al., 2025; Yuan et al., 2025). Proteina-Complexa uses no search. Bold/underline mark the best/second-best reported values.

Type	Method	Des (↑)	DPT (↓)
MLLM	ESM3 (Hayes et al., 2025)	61%	0.37
	DPLM-2 650M (Wang et al., 2025)	63%	0.37
semi-LDM	LSD (Yim et al., 2025)	69%	0.46
LDM	LatentDiff (Fu et al., 2024)	17%	0.51
	PROTEINAE-PLDM (Li et al., 2025)	**93%**	0.36
	Proteina-Complexa (no search) (Didi et al., 2026)	71%	<u>0.20</u>
	PLDesign direct	53%	**0.16**
	PLDesign search	<u>74%</u>	0.29

Table 3: AF3 interface quality across six targets and all available binder lengths. Screen-passing structures receive equal weight within each target, and the resulting target summaries are macro-averaged. Brackets give 95% intervals from 20,000 target–structure hierarchical bootstrap replicates. Bold/underline mark the best/second-best point estimates.

Metric	Dir.	PLDesign $n = 252$	BoltzGen $n = 2{,}583$	PXDesign $n = 4{,}876$	RFDiffusion3 $n = 3{,}341$
Contact pairs	↑	56.677 [51.625, 62.495]	<u>58.080</u> [53.759, 63.915]	**61.914 [55.684, 69.499]**	55.543 [50.065, 61.389]
Contacts / target-interface residue	↑	2.445 [2.367, 2.523]	**2.585 [2.538, 2.637]**	2.448 [2.393, 2.521]	<u>2.462</u> [2.398, 2.528]
Contacts / binder-interface residue	↑	**2.679 [2.601, 2.748]**	<u>2.653</u> [2.578, 2.726]	2.623 [2.548, 2.715]	2.626 [2.543, 2.687]
Contacts / total interface residue	↑	<u>1.269</u> [1.246, 1.291]	**1.299 [1.285, 1.317]**	1.260 [1.241, 1.282]	1.262 [1.245, 1.280]
Compact local score	↑	**1.593 [1.542, 1.643]**	1.560 [1.501, 1.619]	<u>1.575</u> [1.517, 1.640]	1.567 [1.504, 1.615]
Interface quality score	↑	**0.170 [0.155, 0.182]**	0.165 [0.155, 0.173]	<u>0.166</u> [0.159, 0.174]	0.150 [0.141, 0.158]

2023; Team et al., 2025; Zhang et al., 2025). Each metric is averaged over passing candidates within a target, after which the six targets receive equal weight. This prevents targets with more retained candidates from dominating the comparison.

Under this target-macro aggregation, PLDesign has the highest point estimates for contacts per binder-interface residue (2.679), compact local score (1.593), and interface quality score (0.170). The estimates fall within similar ranges across methods, and their overlapping confidence intervals leave the ordering uncertain (Table 3). These contact patterns make retained candidates a potential source of local patches for interface seed development. The measurements concern interfaces after the common sequence-design and AF3 prediction workflow, complementing the raw-coordinate analysis in Appendix L.

Scaffold diversity among retained candidates. Alongside local interface geometry, we examine the range of binder scaffolds represented in the screen-passing cohort. Using binder-only Foldseek comparisons, we summarize structural similarity and cluster diversity within each target, then average the target-specific values with equal weight. PLDesign has a mean nearest-neighbor balanced TM of 0.576, a unique-cluster proportion of 94.0%, and a Hill $q = 1$ effective diversity of 35.823 (Table 4). These results indicate that the retained candidates span varied scaffold geometries, providing multiple structural contexts in which to examine potential interface seeds.

Scaffold reuse across distinct target contexts. To relate the observed scaffold diversity to training structures, we retrieve each binder's nearest training-dimer chain and compare the design target with that chain's associated partners. Among 242 candidates with both comparisons available, 86.4% match a training chain at a Foldseek alignment TM of at least 0.7. Within this subset, 99.0% have a maximum target–partner strict TM below 0.5, indicating that binder similarity usually accompanies a distinct global target context (Table 27). This pattern is consistent with reuse of familiar scaffold geometries beyond the partner configurations associated with the nearest training examples, suggest-

Table 4: Screen-passing binder diversity. Binder-only Foldseek balanced TM is the geometric mean of directional scores. Edges at 0.70 define within-target single-linkage components. Cohort sizes differ across methods and affect diversity estimates. Brackets give 95% target-bootstrap intervals. Bold/underline mark the best/second-best point estimates.

System	Nearest-neighbor TM ↓	Unique-cluster proportion ↑	Hill $q = 1$ effective diversity ↑
PLDesign	**0.576 [0.537, 0.609]**	**94.0% [90.8%, 97.1%]**	**35.823 [16.933, 59.172]**
BoltzGen	0.836 [0.820, 0.852]	<u>17.4% [12.1%, 23.3%]</u>	<u>5.913 [4.194, 7.340]</u>
PXDesign	0.834 [0.816, 0.849]	4.0% [2.5%, 5.4%]	1.390 [1.234, 1.546]
RFDiffusion3	<u>0.817 [0.797, 0.840]</u>	14.6% [10.0%, 20.2%]	4.425 [2.637, 6.583]

ing a basis for exploring new interaction contexts. Appendix L.2 provides the partner definitions, retrieval coverage, and per-target results.

Target variation during joint complex sampling. Joint latent sampling allows the target to vary around its reference structure as the binder is generated. After global target alignment, PLDesign exhibits a target-macro backbone displacement of 0.708 Å within 10 Å of the hotspot. We also track target-side interface residues through reference-templated AF3 prediction. Among candidates with an identifiable raw interface, the target-macro proportion retaining at least 80% of these residues is 17.5%. Local target variation therefore coexists with interface residue retention in a subset of designs, raising the possibility that some sampled arrangements resemble conformational accommodation associated with induced fit. Testing this connection will require experimental binding measurements and structural characterization of the target in its unbound and binder-bound states. Appendix L.1 reports the full motion and retention analyses.

Computational cost of generation and search. On PD-L1, PLDesign requires 11.6 seconds per output with direct sampling and 123.8 seconds with latent search on a single V100 (Table 28). Under the same hardware and beam-width settings, generation with latent search uses approximately 12% of Proteina-Complexa's peak GPU memory and achieves about 3.5 times its output throughput. Appendix M details the measurement protocol.

5 Discussion, Limitations, and Conclusion

PLDesign demonstrates unified autoencoder latents that preserve atomic and interchain geometry, with perturbation analyses connecting the code to structural variation. Trajectory search makes this information usable for inference-time selection, at lower time and memory costs than the tested predictor-guided baseline. The generated candidates further exhibit varied scaffolds and local interaction patches with potential value as interface seeds.

Direct complex generation remains difficult, as reflected in the low computational screening yield. The generator must organize chain conformations, relative placement, and atomic context jointly, and their entanglement within compressed codes may complicate distribution learning. Decoder sensitivity, model capacity, and training coverage are additional possible constraints whose effects remain unresolved. Latent organization is also being investigated as a factor in image generation (Yue et al., 2026).

A promising direction is to distribute geometric responsibilities between explicit coordinates and learned codes. Proteina-Complexa jointly models C_α coordinates and residue-level latents for amino-acid identities and the remaining atomic geometry (Didi et al., 2026). Adapting this partially latent formulation to complete-complex encoding could anchor coarse structural organization while retaining finer, context-dependent information in the latent channels. The current experiments motivate evaluating such representations through both interaction geometry and their ability to support generation and search. By connecting complex-level encoding with generative exploration, PLDesign provides an experimental foundation for investigating how compact representations can support the modeling and design of interacting molecular systems.

AI USE STATEMENT

Generative AI tools assisted with manuscript restructuring, language editing, and code optimization. The authors manually verified all AI-assisted text, numerical values, citations, and scientific claims against the underlying experimental records and source literature. The authors take responsibility for the final content of this work.

ETHICS STATEMENT

PLDesign is a computational method for protein binder design. Such methods can support therapeutic discovery and biological research, while the same capabilities may be misused to design harmful molecular interactions. This study reports computational benchmarks and does not involve human participants, animal experiments, or clinical data.

REPRODUCIBILITY STATEMENT

The main text specifies the model formulation, conditioning mechanism, search procedure, datasets, and evaluation design. The appendices document data processing, architecture and hyperparameters, training and sampling settings, metric definitions, reconstruction ablations, PLDM search experiments, full-grid binder yield, raw and AF3 interface analyses, target-motion measurements, diversity, and structural retrieval. Together these sections provide the configurations and reported measurements needed to reproduce the computational evaluation.

REFERENCES

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024.

Nathaniel R Bennett, Brian Coventry, Inna Goreshnik, Buwei Huang, Aza Allen, Dionne Vafeados, Ying Po Peng, Justas Dauparas, Minkyung Baek, Lance Stewart, et al. Improving de novo protein binder design with deep learning. *Nature Communications*, 14(1):2625, 2023.

Joey Bose, Tara Akhound-Sadegh, Guillaume Huguet, Kilian Fatras, Jarrid Rector-Brooks, Cheng-hao Liu, Andrei Nica, Maksym Korablyov, Michael Bronstein, and Alexander Tong. Se (3)-stochastic flow matching for protein backbone generation. In *International Conference on Learning Representations*, volume 2024, pp. 22590–22621, 2024.

Jasper Butcher, Rohith Krishna, Raktim Mitra, Rafael Isaac Brent, Yanjing Li, Nathaniel Corley, Paul T Kim, Jonathan Funk, Simon Valentin Mathis, Saman Salike, Aiko Muraishi, Helen Eisenach, Tuscan Rock Thompson, Jie Chen, Yuliya Politanska, Enisha Sehgal, Brian Coventry, Odin Zhang, Bo Qiang, Kieran Didi, Maxwell Kazman, Frank DiMaio, and David Baker. De novo design of all-atom biomolecular interactions with rfdiffusion3. *bioRxiv*, 2025. doi: 10.1101/2025.09.18.676967. URL https://www.biorxiv.org/content/early/2025/11/19/2025.09.18.676967.

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, 2026. URL https://arxiv.org/abs/2605.03360.

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.

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.

Rohit Dilip, Evan Zhang, Ayush Varshney, and David Van Valen. Flow autoencoders are effective protein tokenizers. In *International Conference on Learning Representations*, volume 2026, pp. 145901–145926, 2026.

Jennifer Fleming, Paulyna Magana, Sreenath Nair, Maxim Tsenkov, Damian Bertoni, Ivanna Pidruchna, Marcelo Querino Lima Afonso, Adam Midlik, Urmila Paramval, Augustin Žídek, et al. Alphafold protein structure database and 3d-beacons: new data and capabilities. *Journal of molecular biology*, 437(15):168967, 2025.

Cong Fu, Keqiang Yan, Limei Wang, Wing Yee Au, Michael Curtis McThrow, Tao Komikado, Koji Maruhashi, Kanji Uchino, Xiaoning Qian, and Shuiwang Ji. A latent diffusion model for protein structure generation. In *Learning on graphs conference*, pp. 29–1. PMLR, 2024.

Zhangyang Gao, Cheng Tan, and Stan Z. Li. Foldtoken4: Consistent & hierarchical fold language. *bioRxiv*, 2024. doi: 10.1101/2024.08.04.606514. URL https://doi.org/10.1101/2024.08.04.606514.

Tomas Geffner, Kieran Didi, Zuobai Zhang, Danny Reidenbach, Zhonglin Cao, Jason Yim, Mario Geiger, Christian Dallago, Emine Kucukbenli, Arash Vahdat, et al. Proteina: Scaling flow-based protein structure generative models. In *International Conference on Learning Representations*, volume 2025, pp. 98803–98851, 2025.

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.

Thomas Hayes, Roshan Rao, Halil Akin, Nicholas J Sofroniew, Deniz Oktay, Zeming Lin, Robert Verkuil, Vincent Q Tran, Jonathan Deaton, Marius Wiggert, et al. Simulating 500 million years of evolution with a language model. *Science*, 387(6736):850–858, 2025.

Po-Ssu Huang, Scott E Boyken, and David Baker. The coming of age of de novo protein design. *Nature*, 537(7620):320–327, 2016.

John B Ingraham, Max Baranov, Zak Costello, Karl W Barber, Wujie Wang, Ahmed Ismail, Vincent Frappier, Dana M Lord, Christopher Ng-Thow-Hing, Erik R Van Vlack, et al. Illuminating protein space with a programmable generative model. *Nature*, 623(7989):1070–1078, 2023.

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.

Brian Kuhlman and Philip Bradley. Advances in protein structure prediction and design. *Nature reviews molecular cell biology*, 20(11):681–697, 2019.

Alex J Li and Tanja Kortemme. Proteinzen: combining latent and se (3) flow matching for all-atom protein generation. In *Machine Learning for Structural Biology Workshop*, pp. 1–15, 2024.

Shaoning Li, Le Zhuo, Yusong Wang, Mingyu Li, Xinheng He, Fandi Wu, Hongsheng Li, and Pheng-Ann Heng. Proteinae: Protein diffusion autoencoders for structure encoding. *arXiv preprint arXiv:2510.10634*, 2025.

Xiaohan Lin, Zhenyu Chen, Yanheng Li, Zicheng Ma, Chuanliu Fan, Ziqiang Cao, Shihao Feng, Yi Qin Gao, and Jun Zhang. Tokenizing foldable protein structures with machine-learned artificial amino-acid vocabulary. *bioRxiv*, pp. 2023–11, 2023a.

Xiaohan Lin, Zhenyu Chen, Yanheng Li, et al. ProTokens: A machine-learned language for compact and informative encoding of protein 3D structures. *bioRxiv*, pp. 2023.11.27.568722, 2023b.

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.

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

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Amy X Lu, Wilson Yan, Kevin K Yang, Vladimir Gligorijevic, Kyunghyun Cho, Pieter Abbeel, Richard Bonneau, and Nathan C Frey. Tokenized and continuous embedding compressions of protein sequence and structure. *Patterns*, 6(6), 2025.

Mahdi Pourmirzaei, Alex Morehead, Farzaneh Esmaili, Jarett Ren, Mohammadreza Pourmirzaei, and Dong Xu. Gcp-vqvae: A geometry-complete language for protein 3d structure. *bioRxiv*, 2025. doi: 10.1101/2025.10.01.679833. URL https://www.biorxiv.org/content/early/2025/10/03/2025.10.01.679833.

Hannes Stark, Felix Faltings, MinGyu Choi, Yuxin Xie, Eunsu Hur, Timothy John O'Donnell, Anton Bushuiev, Talip Uçar, Saro Passaro, Weian Mao, Mateo Reveiz, Roman Bushuiev, Tomáš Pluskal, Josef Sivic, Karsten Kreis, Arash Vahdat, Shamayeeta Ray, Jonathan T. Goldstein, Andrew Savinov, Jacob A. Hambalek, Anshika Gupta, Diego A. Taquiri-Diaz, Yaotian Zhang, A. Katherine Hatstat, Angelika Arada, Nam Hyeong Kim, Ethel Tackie-Yarboi, Dylan Boselli, Lee Schnaider, Chang C. Liu, Gene-Wei Li, Denes Hnisz, David M. Sabatini, William F. DeGrado, Jeremy Wohlwend, Gabriele Corso, Regina Barzilay, and Tommi Jaakkola. Boltzgen: Toward universal binder design. *bioRxiv*, 2025. doi: 10.1101/2025.11.20.689494.

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.

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

Xinyou Wang, Zaixiang Zheng, Fei Ye, Dongyu Xue, Shujian Huang, and Quanquan Gu. Dplm-2: A multimodal diffusion protein language model. In *International Conference on Learning Representations*, volume 2025, pp. 35463–35490, 2025.

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, Andrew Campbell, Andrew YK Foong, Michael Gastegger, José Jiménez-Luna, Sarah Lewis, Victor Garcia Satorras, Bastiaan S Veeling, Regina Barzilay, Tommi Jaakkola, et al. Fast protein backbone generation with se (3) flow matching. *arXiv preprint arXiv:2310.05297*, 2023.

Jason Yim, Marouane Jaakik, Ge Liu, Jacob Gershon, Karsten Kreis, David Baker, Regina Barzilay, and Tommi Jaakkola. Hierarchical protein backbone generation with latent and structure diffusion. *arXiv preprint arXiv:2504.09374*, 2025.

Mingzhi Yuan, Ao Shen, Kexue Fu, Jiaming Guan, Yingfan Ma, Qin Qiao, and Manning Wang. Proteinmae: masked autoencoder for protein surface self-supervised learning. *Bioinformatics*, 39 (12):btad724, 2023.

Xinyu Yuan, Zichen Wang, Marcus Collins, and Huzefa Rangwala. Protein structure tokenization: Benchmarking and new recipe. *arXiv preprint arXiv:2503.00089*, 2025.

Zhengrong Yue, Taihang Hu, Mengting Chen, Haiyu Zhang, Zihao Pan, Tao Liu, Zikang Wang, Jinsong Lan, Xiaoyong Zhu, Bo Zheng, and Yali Wang. What matters for diffusion-friendly latent manifold? prior-aligned autoencoders for latent diffusion, 2026. URL https://arxiv.org/abs/2605.07915.

Odin Zhang, Xujun Zhang, Haitao Lin, Cheng Tan, Qinghan Wang, Yuanle Mo, Qiantai Feng, Gang Du, Yuntao Yu, Zichang Jin, Ziyi You, Peicong Lin, Yijie Zhang, Yuyang Tao, Shicheng Chen, Jack Xiaoyu Chen, Chenqing Hua, Weibo Zhao, Runze Ma, Yunpeng Xia, Kejun Ying, Jun Li, Yundian Zeng, Lijun Lang, Peichen Pan, Hanqun Cao, Zihao Song, Bo Qiang, Jiaqi Wang, Pengfei Ji, Lei Bai, Jian Zhang, Chang yu Hsieh, Pheng Ann Heng, Siqi Sun, Tingjun Hou, and Shuangjia Zheng. Odesign: A world model for biomolecular interaction design, 2025. URL https://arxiv.org/abs/2510.22304.

TECHNICAL APPENDICES AND SUPPLEMENTARY MATERIAL

The appendices provide dataset construction, representation and architecture details, training and sampling protocols, reconstruction ablations, PLDM generation results, full-grid binder yield, raw and AF3 interface analysis, target-motion measurements, diversity, structural retrieval, and broader impacts.

A DATA PROCESSING PIPELINE

Data sources and preprocessing stages. Our training data combines PDB biological multimers that provide native target-binder interfaces, PDB monomers that preserve fold coverage, and AFDB structures (Fleming et al., 2025) obtained from the clustered and filtered La-Proteina dataset (Geffner et al., 2026), which uses Foldseek-based clustering (Van Kempen et al., 2024). The preprocessing pipeline contains six stages applied to the PDB SeqRes release of 2024-12-26. Table 5 summarizes each stage, its output, and its filtering criteria.

Table 5: Data processing pipeline stages. Each stage applies filters for structural quality, sequence validity, and separation of training and validation groups.

Stage	Operation and output	Key filters and criteria
1. FASTA extraction	Parse PDB SeqRes, output monomer and multimer FASTA files, and perform exact-sequence deduplication.	Discard non-protein chains. Remove sequences shorter than four residues, sequences with non-standard amino acids, and low-complexity sequences. The count changes from 900K to 880K.
2. Sequence clustering	Cluster monomer sequences with MMseqs2 at 40% identity and 80% coverage. Output the cluster mapping.	MMseqs2 removes short or low-complexity sequences and logs remaining unclustered sequences. The result contains 41,456 clusters with 18,475 multi-member clusters and 22,981 singletons.
3. HDF5 construction	Parse mmCIF files, extract coordinates, compute interface residues, and write HDF5 and index files.	Reject crystal-only assemblies. Require at least 50% CA completeness. Truncate at 20 chains or 5120 tokens. Detect CA breaks above 10 Å. Remove entries with more than 30% clashing atoms. Use a 5 Å interface cutoff. The result contains 846K interface records and 719K chain records.
4. Cluster ID correction	Replace raw chain IDs with MMseqs2 cluster representatives and recompute sampling weights with $w = 1/c$.	Remove records with unmappable cluster IDs. Interface records change from 846K to 35K across 16,434 clusters. Monomer records change from 100K to 44K across 16,890 clusters.
5. Training and validation split	Apply Union-Find to the cluster graph, use a date cutoff, and select validation records by group.	Use release dates through 2021-09-30. Require resolution below 3.0 Å for validation groups and set val_split_ratio to 0.001. The result contains 62,231 records in 14,554 connected groups.
6. Merge and finalize	Concatenate training and validation TSVs, recompute merged weights, and produce final index files.	The final split contains **62,086 training records**, including 38,588 monomers and 23,498 interfaces, plus 145 validation records.

Training set composition and estimated coverage. The PDB training partition contains two record types drawn from the same pool of experimental structures. The AFDB component is supplied separately by the clustered and filtered La-Proteina dataset. Table 6 reports the composition statistics and the estimated number of training steps required to achieve k-representative coverage at 99% confidence under weighted sampling with $w = 1/c$. Specifically, a cluster of size c is considered covered once at least $k(c) = \lfloor \log_2 c \rfloor + 1$ distinct members have been sampled. This k-

representative criterion samples multiple conformations from large, abundant protein families while requiring fewer steps than full sample-wise coverage.

Table 6: Final training set composition and k-representative coverage estimates. The step estimates use binary-search numerical integration of the weighted Coupon Collector problem. Both record types reach a full-coverage probability of 1.0 at the reported step counts.

Record type	**Records**	**Clusters**	**Mean c**	**Max c**	**k-rep. steps**
Monomer chains	38,588	15,084	2.6	358	429,633
Interface pairs	23,498	11,666	2.0	1,372	389,684
Combined total	62,086				

Interface residue definition. An interface residue is defined as any residue whose heavy atoms fall within a distance cutoff of 5.0 Å of any heavy atom belonging to a different chain within the same biological assembly. The detection algorithm uses cKDTree spatial queries with bounding-box pre-filtering for efficiency. Residues participating in at least one interface are classified as interface chains, while residues not participating in any interface are classified as non-interface monomers. This binary classification determines whether a sample contributes to the monomer or the interface partition of the training mixture.

Sampling weights and data mixture. Each training record is assigned a sampling weight $w_i = 1/c_i$, where c_i is the size of its sequence cluster. This gives each cluster approximately equal total sampling probability. At every step, one record is drawn in proportion to w_i and cropped to 896 residues for autoencoder training or 448 residues for PLDM training. Uniform $\mathrm{SO}(3)$ rotation augments the coordinate frame. The effective sample size is approximately 52% of the nominal monomer partition and 60% of the interface partition.

CASP benchmark processing. The CASP14 and CASP15 dimer lists were constructed once and frozen as reconstruction evaluation sets. They remain outside gradient-based optimization and receive no model-specific filtering. The model configurations and Epoch 99 endpoint were fixed independently of their reconstruction metrics. We retain complexes with at least one inter-chain heavy-atom contact within 5 Å, remove complexes longer than 2,048 residues, and exclude assemblies with more than nine chains when their total length exceeds 1,536 residues. Every checkpoint receives the identical retained complexes, chain order, coordinate centring, and valid-atom masks. Each model produces one reconstruction per complex, followed by the same Kabsch alignment, RMSD implementation, and DockQ implementation. No model-specific best-of-k selection is used. ProteinAE and PLDesign retain the same DiT width and encoder/decoder depth and use the same $\Delta t = 0.05$ decoder ODE integration, fixing the continuous-AE reconstruction budget at 20 steps. FoldToken-4 outputs are mapped to the shared $\{\mathrm{N}, \mathrm{C}_\alpha, \mathrm{C}, \mathrm{O}\}$ atom selection before the common alignment and metric calculation.

B Representation and Masking Details

Atom14 tensor layout. A protein complex of length $L = L_T + L_B$ is represented as the stacked tensor $x = (x_T, x_B) \in \mathbb{R}^{L\times 14\times 3}$, where the 14 atom slots per residue cover backbone and heavy atoms. Residues lacking specific heavy atoms are padded with a dedicated dummy token `DMY`. Per-residue side fields include the residue type $r \in \{0, \ldots, 20\}^L$, the chain index $c \in \{0, \ldots, 7\}^L$, and the target hotspot indicator $h \in \{0, 1\}^L$. All coordinates are expressed in nanometres.

Slot ordering, dummy handling and masking. In the Atom14 complex representation, positions $\{1, 2, 3, 4\}$ are always the backbone atoms $\{\mathrm{N}, \mathrm{C}_\alpha, \mathrm{C}, \mathrm{O}\}$ and positions $\{5, \ldots, 14\}$ are residue-specific heavy atoms following the canonical named-slot placement used by A-CODE and the side-chain ordering used by BoltzGen (Cheng et al., 2026; Stark et al., 2025). Our dummy token `DMY` is an independent symbol in the atom-name vocabulary. This gives the model a distinct channel for

inapplicable atoms and supports a uniform 14-slot tensor across residue types. The network input mask covers all 14 slots of every valid residue uniformly. A separate ground-truth mask $m_{\mathrm{gt}} \in \{0,1\}^{L\times 14}$, constructed from the canonical per-residue atom table and the PDB presence mask, is used only for loss computation. Dummy slots receive a data-dependent initialisation $x_{i,a} = x_{i,\mathrm{C}_\alpha} + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma^2_{\mathrm{DMY}} I)$ and $\sigma_{\mathrm{DMY}} = 0.05\,\mathrm{nm}$, aligning their flow trajectories with real side-chain atoms.

C Architecture Details

The autoencoder and PLDM share the AlphaFold3-style Diffusion Transformer formulation used by ProteinAE (Li et al., 2025). The PLDesign AE preserves ProteinAE's trunk width, pair width, attention-head count, and five-layer encoder/five-layer decoder depth. The representation, chain/interface inputs, and Atom14 output define the complex extension. The model maintains a per-residue representation s and a pair representation p. The pair representation acts as an additive attention bias and does not use a stand-alone pair track.

Interface Conditioning Features (ICF). The ICF module generalizes hotspot conditioning to complete complexes and carries cross-chain context throughout the model. Given the hotspot field h, chain index c, and reference C_α geometry, it constructs a layered contact map $C \in \mathbb{R}^{L\times L}$. Values in $[-1, 2]$ distinguish non-interface, interface-proximity, and binder-binder states. A two-layer projection adds the continuous map to the pair representation p through a zero-initialized scalar gate α_{ICF}.

$$p \leftarrow p + \alpha_{\mathrm{ICF}} \cdot \mathrm{LayerNorm}\big(W_2\,\mathrm{ReLU}(W_1\,C)\big), \qquad W_1 \in \mathbb{R}^{128\times 1}, \qquad W_2 \in \mathbb{R}^{128\times 128}. \quad (3)$$

The same ICF bias is shared across the encoder, PLDM, and decoder, keeping interface context consistent throughout the generative pipeline. ICF acts as an interface-oriented feature extractor. Its layered pair map directs representational capacity toward cross-chain neighborhoods and target–binder proximity. The corresponding ablation measures the efficiency of this interface-aware feature extraction under a fixed complex-AE architecture.

DiT block. Each DiT layer takes s, p, and a conditioning vector c and applies adaptive LayerNorm (AdaLN), pair-biased multi-head attention with QK-LayerNorm and rotary positional encoding, an AdaLN output scale, and a conditioned transition. The initial s^0 concatenates embeddings of the chain index, the residue-position index, and any module-specific features. The initial p^0 concatenates a relative-sequence-separation embedding with module-specific binned distances. Learnable registers are appended to s, following the AF3 practice.

Encoder and decoder specialisations. The three networks share the same AF3-style DiT backbone and differ only in the feature lists they consume. The AE encoder E_ψ takes the clean complex x_1 as input. Sequence features include the chain index. Pair features include relative sequence separation and binned distances derived from x_1. There is no time embedding, and triangular multiplication is disabled. A per-residue representation is projected channel-wise to the latent dimension by a single linear map, yielding $z \in \mathbb{R}^{L\times 16}$. No length downsampling is applied. The latent preserves per-residue addressability for downstream interface-aware conditioning and search.

The AE decoder D_ϕ takes the noised complex x_t, the diffusion time t, and the latent z as input. Sequence features include the chain index and a self-conditioning stub. Pair features also include binned distances derived from x_t and the self-conditioning estimate. Triangular multiplication is enabled on the pair track, since reconstructing interface geometry benefits from explicit triangle consistency. The final head outputs the flow velocity over Atom14 positions.

Following AF3, a small atom-transformer aggregates the 14 atoms of each residue into a token representation before the DiT trunk and broadcasts the trunk output back to the atom level at the end.

Normalisation of the latent. We follow the LayerNorm-style normalisation used in ProteinAE and do not regularise z with a KL term. A LayerNorm-like normalisation inside the encoder projection prevents the latent from drifting without requiring a scale hyperparameter. The downstream

latent flow then consumes z directly, with no further rescaling. This keeps the two training stages decoupled at the interface level.

Latent denoiser trunk. The latent trunk is again an AF3-style DiT. Triangular multiplication is disabled at the latent stage, because without physical coordinates the triangle-consistency prior has no geometric meaning. The pair track carries only relative sequence separation plus the ICF and template residuals. The 16-dimensional AE latent is projected into the trunk width through a linear layer and enters as a sequence-level self-conditioning feature.

AF3-style target template embedding. The template contributes six feature groups computed from its backbone atoms $(x^T_{\mathrm{N}}, x^T_{C_\alpha}, x^T_{\mathrm{C}}, x^T_{C_\beta})$. These include a pseudo-β distogram, a pseudo-β pair mask, three unit-direction components obtained by expressing inter-residue displacements in the local backbone frame, a backbone pair mask, and a query embedding that carries the current pair context. Each group is linearly projected and summed, and the ICF contact map is optionally fused as a eighth group. The summed activation is refined by two lightweight PairFormer-style layers combining triangular multiplication and a pair transition, and the refined tensor is mapped back to the pair dimension through a zero-initialised linear layer.

ICF projection. The ICF projection is a two-layer MLP $\mathbb{R} \to \mathbb{R}^{128} \to \mathbb{R}^{128}$ fused into the pair bias as a 128-dimensional additive term. Simple ICF uses the value range $[-1, 2]$ with three semantic states (non-interface, interface proximity, binder-binder region). Simple ICF (selected model) exhibits clear block-diagonal structure from intra-chain distance geometry, with concentrated off-diagonal bands corresponding to inter-chain contacts at the binding interface. No ICF retains off-diagonal structure but with higher background noise, indicating that cross-chain coupling is partially recovered from coordinate-based pair features alone. Joint centre-of-mass centring without interface features (cc_nic) produces the weakest but still non-trivial off-diagonal signal (Figure 5).

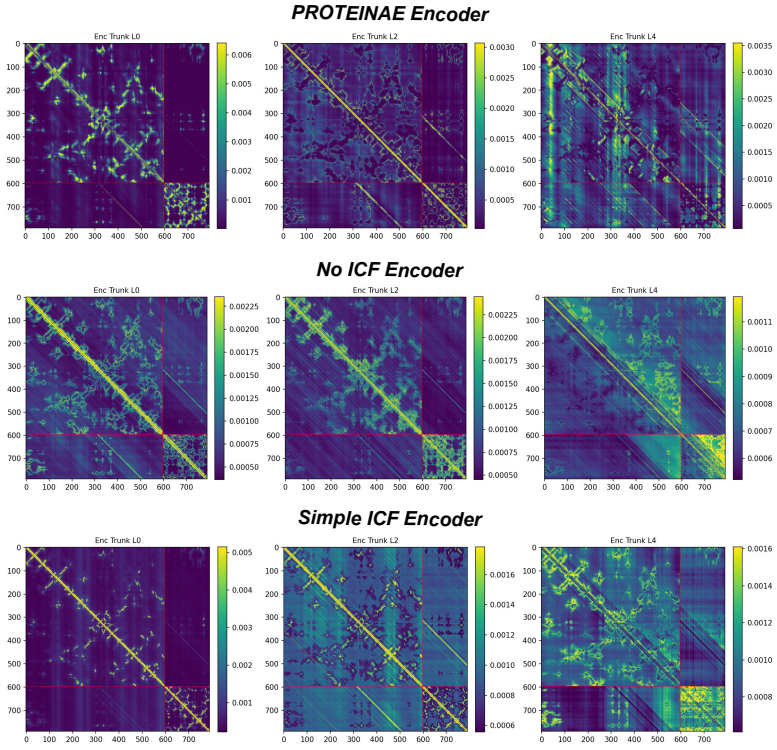

Figure 5: Encoder pair-attention visualisation on a representative dimer. Encoder attention maps for ProteinAE and PLDesign variants trained with different ICF modules (No ICF and Simple ICF) are shown from top to bottom. The colour scale indicates the attention weight of each pair.

D Model Architecture Hyperparameters

Table 7 summarizes the architectural hyperparameters for the autoencoder and PLDM.

Table 7: Architectural hyperparameters for the autoencoder and PLDM.

Hyperparameter	**Autoencoder**	**PLDM**
Number of layers	5 (Encoder) + 5 (Decoder)	15
Token dimension (D_s)	256	768
Pair dimension (D_p)	128	128
Attention heads	8	12
Latent dimension (d)	16	-
Conditioning dimension	-	512
Learnable registers	32	32
Atom-transformer layers	3	-
Atom-transformer channels	64	-

E Flow-Matching Preliminaries and Coordinate Preprocessing

Rectified flow-matching preliminaries. We adopt rectified flow matching (Lipman et al., 2022; Liu et al., 2022) for both the autoencoder decoder and PLDM. Given a data sample x_1 and a Gaussian reference $x_0 \sim \mathcal{N}(0, I)$, we construct the linear interpolant $x_t = (1-t)x_0 + tx_1$ for $t \in [0, 1]$. The coordinate-space decoder restricts the reference distribution to the centred subspace, whereas the latent-space PLDM operates without coordinate centring. Sampling proceeds from $t = 0$ to $t = 1$, moving from the Gaussian reference toward the data distribution. Accordingly, larger t corresponds to a smaller noise coefficient $1 - t$. The interpolation induces the conditional target velocity $u(x_t \mid x_0, x_1) = x_1 - x_0$. A neural vector field $v_\theta(x_t, t, c_{\text{cond}})$ is trained using the conditional flow-matching objective.

Joint zero centre of mass. At the coordinate level, we apply a joint zero centre-of-mass strategy for both complex and monomer samples, centering atomic coordinates around the origin by the joint C_α centroid. This aligns the atomic distributions of complexes and monomers around a common origin and makes it easier for the flow field to learn atomic distribution patterns across the mixed dataset. Unlike target-centric training strategies that require two-stage curricula, our joint-CoM preprocessing avoids the negative impact of distribution discrepancies between monomer and complex data and enables single-stage AE training on the mixture.

All-atom complex AE training. The autoencoder is trained jointly on PDB dimers and monomers in the Atom14 representation, so that the induced latent $z = E_\psi(x)$ retains all-atom interface fidelity. The reconstruction objective combines a ground-truth-masked flow-matching loss with stereochemistry-aware auxiliary terms:

$$\mathcal{L}_{\text{AE}} = \mathbb{E}_{t,x_0,x_1}\left[\frac{\left\|\big(v_\theta(x_t,t,z) - (x_1 - x_0)\big) \odot m_{\text{gt}}\right\|^2}{(1-t_\star)^2}\right] + \lambda_{\text{bond}}\mathcal{L}_{\text{bond}} + \lambda_{\text{lddt}}\mathcal{L}_{\text{lddt}} + \lambda_{\text{dgm}}\mathcal{L}_{\text{dgm}}, \tag{4}$$

where $m_{\text{gt}} \in \{0,1\}^{L\times 14}$ is the ground-truth atom mask (dummy slots are excluded from supervision) and $t_\star = \min(t, 0.9)$ is a stability clamp. To ensure that PLDesign remains usable with only partial hotspot labels at inference time, we additionally apply a random masking strategy to the ICF contact map during AE training, independently masking each residue row with probability 0.3. Activation schedules of the auxiliary losses, data augmentation, and coordinate preprocessing are described in Appendices E and F.

Auxiliary-loss activation schedule and coordinate augmentation. In Eq. (4), the auxiliary terms $\mathcal{L}_{\text{bond}}$, $\mathcal{L}_{\text{lddt}}$, and $\mathcal{L}_{\text{dgm}}$ are activated only for $t > \tau_{\text{AE}}$, where τ_{AE} defines the start of the low-

noise regime. This time gate concentrates geometric supervision near the data end of the interpolation, where the noise coefficient $1 - t$ is small. The main reconstruction loss is evaluated only at valid ground-truth atom positions selected by m_{gt}. Dummy slots participate in the neural trajectory through x_t but are excluded from loss computation. Coordinate-space augmentation applies a uniformly sampled $\mathrm{SO}(3)$ rotation to the entire complex at each training step.

F Training Details

Latent endpoint objective. We optimize the time-weighted endpoint loss, normalizing each chain by its valid residue count.

$$\mathcal{L}_{\mathrm{PLDM}} = \mathbb{E}_{t,z_0,z_1}\left[\frac{w_T \ell_T + w_B \ell_B}{(1-t_\star)^2 + \epsilon}\right], \qquad \ell_C = \frac{1}{d\,|C|}\sum_{i\in C} \|\hat{z}_{1,i} - z_{1,i}\|_2^2, \quad C \in \{T, B\}. \tag{5}$$

Here, T and B contain valid target and binder residues. We set $w_T = w_B = 1$, $t_\star = \min(t, 0.9)$, and $\epsilon = 10^{-5}$. This normalization gives both chains equal weight despite differences in length.

Training objectives and noise schedule. Training uses a shared scalar time for the complete complex, sampled from $t \sim 0.02 \cdot \mathrm{Uniform}(0,1) + 0.98 \cdot \mathrm{Beta}(1.9, 1.0)$ (Geffner et al., 2025). A loss clamp and per-sample outlier clamp stabilize the PLDM. Whole-sample ICF dropout with probability 0.2 provides classifier-free unconditional examples. Per-residue ICF masking at 0.3 simulates partial hotspot knowledge. Uniform $\mathrm{SO}(3)$ rotation augments the coordinate frame. Self-conditioning is used with probability 0.5 by detaching and reusing a one-step rollout of v_φ.

Staged training pipeline. The autoencoder was trained first on PDB dimers, PDB monomers, and Foldseek-clustered AFDB structures with sampling weights of 40%, 10%, and 50% using Eq. (4), AdamW with $\mathrm{lr} = 10^{-4}$, bf16 mixed precision, and EMA decay 0.999. Its encoder and decoder remained frozen throughout PLDM training. The Epoch 99 reporting point was fixed before CASP evaluation. We determined the training horizon by converting the precomputed sampling estimate for at least one expected draw per training record into complete epochs and rounding upward. PLDM training then proceeded in two stages on the same mixture at a crop length of 448. PLDM-Stage 1 optimized the latent flow-matching objective in Eq. (5). PLDM-Stage 2 started from the PLDM-Stage 1 EMA checkpoint and activated decode-space supervision, followed by low-noise bond and smooth-LDDT refinement. This continuation produced the PLDM checkpoint used for the main generation results. PLDM-Stage 1 ran for 150K steps using only the latent MSE loss. PLDM-Stage 2 ran for 35K steps with `decode_loss` enabled. The `bond_loss` and `smooth_lddt_loss` terms were additionally enabled during its final 1K steps.

Decode-aware continuation. The second stage connects latent prediction to the geometry recovered by the decoder. Starting from the first-stage EMA checkpoint, we retained $\mathcal{L}_{\mathrm{PLDM}}$ and added supervision on decoded coordinates. The predicted clean latent $\hat{z}_1$ conditions the frozen decoder. For this auxiliary decoder pass, training coordinates are perturbed at the same time t, with target coordinates kept clean when available. The decoder then produces a clean-coordinate estimate $\hat{x}_1$. We measure its deviation from the reference structure using a masked coordinate MSE.

$$\mathcal{L}_{\mathrm{coord}} = \mathbb{E}\left[\frac{\sum_{i=1}^{L}\sum_{a=1}^{14} m_{i,a}^{\mathrm{gt}}\, \|\hat{x}_{1,i,a} - x_{1,i,a}\|_2^2}{3\sum_{i=1}^{L}\sum_{a=1}^{14} m_{i,a}^{\mathrm{gt}}}\right]. \tag{6}$$

The mask m^{gt} excludes missing atoms and dummy slots. Gradients pass through the frozen decoder to the PLDM, directly linking latent predictions to coordinate reconstruction. The encoder and decoder parameters remain unchanged.

During the final 1K steps of this continuation, we added bond and smooth-LDDT losses to refine local geometry. The bond loss penalizes errors in consecutive C_α distances within each chain. The smooth-LDDT loss measures agreement between local C_α distance neighborhoods in the prediction and reference. Both terms are evaluated on low-noise samples with $t > \tau_{\mathrm{aux}}$, where τ_{aux} is the activation threshold. With $\mathcal{L}_{\mathrm{bond}}$ and $\mathcal{L}_{\mathrm{lddt}}$ denoting these noise-gated losses, the second-stage objective is

$$\mathcal{L}_{\mathrm{PLDM}}^{(2)} = \mathcal{L}_{\mathrm{PLDM}} + \lambda_{\mathrm{decode}}\left(\mathcal{L}_{\mathrm{coord}} + \lambda_{\mathrm{bond}}\mathcal{L}_{\mathrm{bond}} + \lambda_{\mathrm{lddt}}\mathcal{L}_{\mathrm{lddt}}\right). \tag{7}$$

The coefficient λ_{decode} controls decoded-structure supervision and corresponds to `decode_loss_weight`. The weights λ_{bond} and λ_{lddt} were zero during the first 34K steps of the second stage. They became positive during the final 1K steps, while the coordinate MSE and latent flow-matching losses remained active. The resulting checkpoint is used for subsequent generation and latent trajectory search.

All PLDM results reported in this paper use the final checkpoint obtained after this complete training schedule.

The AE has $\sim 40\text{M}$ parameters. The latent trunk has $\sim 200\text{M}$ parameters.

The autoencoder was trained for 155K steps on eight NVIDIA H20 GPUs, each with 96 GB of memory, using a batch size of eight per GPU. The PLDM was trained on 32 NVIDIA A100 GPUs, each with 40 GB of memory, using a batch size of four per GPU. Testing and inference used a pool of eight NVIDIA V100 GPUs, each with 32 GB of memory, while efficiency measurements were conducted on a single GPU.

G Sampling Details

The stochastic sampler augments the latent velocity with a score drift and diffusion,

$$\mathrm{d}z_t = v_\varphi(z_t, t, c_{\text{cond}})\,\mathrm{d}t + g(t)\,\eta_s\, s_\varphi(z_t, t, c_{\text{cond}})\,\mathrm{d}t + \sqrt{2\,g(t)\,\eta_n}\,\mathrm{d}W_t, \tag{8}$$

where s_φ is the conditional score associated with the velocity field. For the linear Gaussian interpolant and $0 < t < 1$, it is given by

$$s_\varphi(z_t, t, c_{\text{cond}}) = \frac{t\,v_\varphi(z_t, t, c_{\text{cond}}) - z_t}{1 - t}. \tag{9}$$

The marginal-preserving form uses the same coefficient $a(t) \geq 0$ in the score drift and diffusion,

$$\mathrm{d}z_t = \big[v_\varphi(z_t, t, c_{\text{cond}}) + a(t) s_\varphi(z_t, t, c_{\text{cond}})\big]\,\mathrm{d}t + \sqrt{2a(t)}\,\mathrm{d}W_t. \tag{10}$$

With an exact velocity and score, Eq. (8) satisfies this condition when $\eta_s = \eta_n$. The reported sampler uses unequal scales to adjust exploration and therefore does not satisfy this marginal-preservation condition. The score expression applies away from $t = 1$, where it is singular. For latent sampling, $g(t) = \min\big(5, (1-t)/t\big)$ follows the optimal-transport schedule. We integrate the SDE with Euler-Maruyama from $t = 0$ to $t = 1$ with default step size $\Delta t = 0.0025$ and default scales $(\eta_s, \eta_n) = (1.0,\, 0.2)$. The sampler can switch to pure ODE integration for the final steps to remove late-stage stochastic perturbations without retraining. For latent-to-coordinate decoding, we integrate the coordinate-space ODE $\mathrm{d}x_t = v_\theta(x_t, t, \hat{z})\,\mathrm{d}t$ with step size $\Delta t = 0.05$ starting from a Gaussian sample in the centred subspace. For de novo generation, the decoded output is retained as a backbone seed. ProteinMPNN assigns residue identities, and AF3 produces the final all-atom complex used for evaluation.

H Evaluation Metrics

H.1 Unconditional Generation Metric Definitions

Evaluation configuration. The unconditional generation assessment contains 690 backbones with ten samples at each integer length from 60 to 128 residues.

Designability. We operationally define designability by sequence–structure self-consistency under ProteinMPNN and ESMFold. ProteinMPNN (Dauparas et al., 2022) designs eight sequences per backbone at temperature 0.1, and ESMFold (Lin et al., 2022) predicts their structures. A backbone is designable when its minimum self-consistency RMSD (scRMSD) is at most 2 Å. The final metric represents the percentage of designable backbones across the entire set.

Designable Pairwise TM-score (DPT). DPT serves as an indicator of structural diversity by measuring the topological similarity among generated samples. We compute the average pairwise TM-score exclusively among designable backbones of the same length and then average the 69 within-length values with equal weight. Consequently, a lower DPT value corresponds to a more diverse structural distribution.

H.2 BINDER GENERATION METRIC DEFINITIONS

Baseline generation pipelines. Every generator receives the same prepared target coordinates, chain crop, hotspot residue set, and requested binder length for each target–length cell. BoltzGen uses its public target-conditioned `protein-anything` protocol to sample all-atom target–binder structures (Stark et al., 2025). PXDesign uses the generation-only PXDesign-d path (`pxdesign infer`) with the target crop, hotspots, and binder length supplied in its task configuration (Team et al., 2025). RFDiffusion3 uses its public protein-binder recipe with the same target, hotspot, and length conditions to sample all-atom complexes (Butcher et al., 2025). Each method is requested to produce 100 raw candidates per cell. Generation failures remain in the planned denominator, and model-native sequence design, predictor ranking, and filtering are omitted so that all raw structures enter one common post-generation workflow.

Common post-generation pipeline. For each processed backbone, ProteinMPNN designs one sequence with shared parameters (Dauparas et al., 2022). ProteinMPNN sequence design followed by structure-prediction validation is a standard two-stage binder-design workflow (Bennett et al., 2023; Team et al., 2025). AlphaFold3 (Abramson et al., 2024) predicts each designed sequence as a complex using the original reference target structure as the target-chain template. The binder chain receives no structural template. PLDesign and all baselines use the same ProteinMPNN and AF3 parameters. Missing, unprocessed, and failed candidates remain in the planned denominator. Metrics are rounded to two decimals before filtering.

AF3 inter-chain confidence. Let p_{ijb} be the AF3 predicted-aligned-error probability for token pair (i, j) in error bin b, and let e_b be the bin centre. The expected aligned error and its TM kernel are

$$\widehat{e}_{ij} = \sum_b p_{ijb} e_b, \qquad q_{ij} = \sum_b p_{ijb} \left[1 + (e_b/d_0)^2\right]^{-1}, \tag{11}$$

where d_0 is the AF3 length-dependent TM normalization. With chain identity $\chi(i)$ and valid-pair weights w_{ij}, ipTM masks same-chain pairs and selects the highest cross-chain alignment-anchor score,

$$\mathrm{ipTM} = \max_i \frac{\sum_{j:\chi(j)\neq\chi(i)} w_{ij} q_{ij}}{\sum_{j:\chi(j)\neq\chi(i)} w_{ij}}. \tag{12}$$

The pTM score applies the same anchor maximization to all valid token pairs and summarizes global complex consistency. For target tokens T and binder tokens B, the symmetrized inter-chain mean PAE used for screening is

$$\mathrm{iPAE}_{\mathrm{mean}} = \frac{1}{|T||B|} \sum_{i\in T, j\in B} \frac{\widehat{e}_{ij} + \widehat{e}_{ji}}{2}. \tag{13}$$

These cross-chain aggregates evaluate relative chain placement across the complete target and binder. Global scaffold geometry and extended regions influence ipTM and iPAE alongside the local interface. The contact-derived metrics below resolve local contact density, compactness, and packing within the interacting patch.

Following established computational protocols (Zhang et al., 2025), a candidate passes the computational screen when

$$\mathrm{pLDDT} > 80, \qquad \mathrm{ipTM} > 0.5, \qquad \mathrm{iPAE} \leq 10.85\ \text{Å}, \qquad \mathrm{RMSD} < 2.5\ \text{Å}. \tag{14}$$

RMSD is the minimum multimer-mode USalign complex RMSD between the generated candidate complex and its AF3 prediction. The 2.5 Å threshold measures reproducibility of the target–binder relative orientation after sequence design. ProteinMPNN and AF3 operate in the common evaluation workflow and do not enter PLDesign training, latent ranking, or seed selection.

Screen-passing cohort diversity. For each target, we compare binder-only structures with Foldseek in both directions and define balanced TM as $\sqrt{\mathrm{qTM}\,\mathrm{tTM}}$. An edge connects two candidates when balanced TM is at least 0.70, and single-linkage connected components define structural families. The nearest-neighbor statistic averages each candidate's largest balanced TM to another screen-passing candidate. The unique-cluster proportion is the number of components divided by

the number of screen-passing candidates, and Hill $q = 1$ is the exponential Shannon diversity of component frequencies. We compute each metric within target and average the six targets equally. Confidence intervals are estimated by resampling targets 200,000 times.

Benchmark targets. Table 8 lists the target configurations used across our binder studies. The current full-grid comparison uses BHRF1, SC2RBD, IL-7RA, PD-L1, TrkA, and InsulinR. These six targets provide the structural context, epitope annotations, and common configurations needed for cross-method comparison across all available binder lengths.

Table 8: Protein targets used in the full-grid binder comparison. Target context lists retained chain identifiers and residue ranges. Hotspots provide interface conditions. Length gives the evaluated binder range at five-residue intervals.

Target	PDB	Target context	Hotspots	Partner	Length
BHRF1	2wh6	A residues 2–158	A65, A74, A77, A82, A85, A93	BH3 helix	80–120
SC2RBD	6m0j	E residues 333–526	E485, E489, E494, E500, E505	ACE2 receptor	80–120
IL-7RA	3di3	B residues 17–209	B58, B80, B139	IL-7	50–120
PD-L1	5o45	A residues 17–132	A56, A115, A123	PD-1	50–120
TrkA	1www	X residues 282–382	X294, X296, X333	Nerve growth factor	50–120
InsulinR	4zxb	E residues 6–155	E64, E88, E96	Insulin receptor	40–120

Interface metrics. Let T be the set of target residues, B the set of binder residues, and $A(r)$ the heavy atoms of residue r. For residues $i \in T$ and $j \in B$, the minimum heavy-atom distance is

$$d_{ij} = \min_{a \in A(i),\, b \in A(j)} \|x_a - x_b\|_2.$$

We use a contact cutoff $\tau_c = 5$Å, a close-contact cutoff $\tau_{\text{close}} = 4$Å, and a clash cutoff $\tau_{\text{clash}} = 2$Å. The contact, close-contact, and clash sets are defined as

$$C = \{(i,j) \mid d_{ij} \le \tau_c\}, \quad C_{\text{close}} = \{(i,j) \mid d_{ij} \le \tau_{\text{close}}\}, \quad K = \{(i,j) \mid d_{ij} \le \tau_{\text{clash}}\}.$$

The target-side and binder-side interface residue sets are

$$I_T = \{i \mid \exists j, (i,j) \in C\}, \qquad I_B = \{j \mid \exists i, (i,j) \in C\},$$

and the mean contact distance is $\bar{d}_C = \frac{1}{\max(|C|,1)} \sum_{(i,j)\in C} d_{ij}$. The contact-derived interface metrics reported in the main paper are computed from these sets. Contact-size metrics quantify the physical extent of the interface. Length- and residue-normalized metrics quantify how efficiently a binder converts its sequence budget into contacts. Local-density metrics measure whether contacts are concentrated on a few residues or distributed more diffusely. Clash-based metrics penalize sterically infeasible geometry. The interface PAE metric averages symmetric PAE only over residue pairs in the contact set C. It is distinct from the full cross-chain block mean iPAE used for screening above. Table 9 gives the exact contact-derived formulas used in Tables 3 and 24.

The *Compact local score* multiplies close-contact enrichment by binder-side packing density, rewarding interfaces that are both tight and locally dense. The *Interface quality score* combines mean-distance proximity, length efficiency, and a clash penalty into a scalar in $[0, 1]$.

I Reconstruction Benchmark

I.1 Latent capacity and multichain training

Table 11 compares three Atom14 complex AEs at the pre-specified Epoch 99 endpoint with the public ProteinAE checkpoint on the same frozen dimer benchmarks. Every complex-trained variant improves substantially over ProteinAE at C_α and backbone resolution. Within the complex AE family, D32 gives the strongest backbone geometry and D16 gives the strongest all-atom reconstruction. Multichain training establishes a representation of relative chain placement, while latent width

Table 9: Contact-derived interface metric formulas and preferred directions. All quantities are computed from the contact set C, close-contact set C_{close}, clash set K, and the target/binder interface residue sets I_T, I_B defined above. For each row, ↑ indicates that larger values are preferred, while ↓ indicates that smaller values are preferred.

Metric	Dir.	Formula
Contact-size metrics		
Contact pairs	↑	$\lvert C\rvert$
Close-contact pairs	↑	$\lvert C_{\text{close}}\rvert$
Length- and residue-normalized efficiency		
Contacts / binder length	↑	$\lvert C\rvert/\lvert B\rvert$
Contacts / target-interface	↑	$\lvert C\rvert/\max(\lvert I_T\rvert, 1)$
Contacts / binder-interface	↑	$\lvert C\rvert/\max(\lvert I_B\rvert, 1)$
Contacts / total interface residue	↑	$\lvert C\rvert/\max(\lvert I_T\rvert + \lvert I_B\rvert, 1)$
Local density and compactness		
Compact local score	↑	$\frac{\lvert C_{\text{close}}\rvert}{\max(\lvert C\rvert,1)} \cdot \frac{\lvert C\rvert}{\max(\lvert I_B\rvert,1)}$
Interface quality score	↑	$\left(1 - \min(\bar{d}_C/\tau_c, 1)\right) \min(\lvert C\rvert/\lvert B\rvert, 1) \left(1 - \min(\lvert K\rvert/\max(\lvert C\rvert, 1), 1)\right)$
AF3 contact confidence		
Interface PAE	↓	$\frac{1}{\max(\lvert C\rvert, 1)} \sum_{(i,j)\in C} \frac{\text{PAE}_{ij} + \text{PAE}_{ji}}{2}$
Steric plausibility		
Clash count	↓	$\lvert K\rvert$
Clashes / contact	↓	$\lvert K\rvert/\max(\lvert C\rvert, 1)$

Notation. $\lvert C\rvert$ and $\lvert C_{\text{close}}\rvert$ count contact and close-contact residue pairs. $\lvert B\rvert$ denotes binder length. $\lvert I_T\rvert$ and $\lvert I_B\rvert$ count target-side and binder-side interface residues. $\lvert K\rvert$ is the clash count. $\bar{d}_C$ is the mean pairwise distance over C, and τ_c is the contact-distance scale used to normalize it. The interface quality score is bounded in $[0, 1]$.

controls how the finite per-residue code distributes capacity between backbone precision and side-chain detail. The downstream PLDM uses the predetermined Atom14 D16 bottleneck. This variant records the lowest all-atom RMSD within the evaluated complex-AE family on both dimer sets.

I.2 Training duration and resolution balance

Table 12 reports a post hoc checkpoint profile around the pre-specified Epoch 99 endpoint. D8 and D16 reach their lowest observed dimer C_α/backbone values at Epochs 32–80, while their all-atom RMSD continues improving through Epoch 99. These retrospective minima describe training dynamics. All main reconstruction comparisons use the pre-specified Epoch 99 endpoint.

I.3 Backbone and all-atom supervision

Table 13 isolates the atom-level objective while retaining the D16 bottleneck. Atom14 D16 shares its code across backbone and side-chain coordinates, which raises its C_α error relative to the backbone-only objective. Backbone D16 removes side-chain supervision and achieves the best C_α RMSD on all five monomer splits, including lower values than ProteinAE. The ablation identifies competition for finite latent capacity and shows that the complex AE architecture can attain leading backbone reconstruction when trained on the same atom set as ProteinAE.

I.4 Interface feature extraction

Tables 14 and 15 evaluate ICF as an interface-oriented feature extractor under a fixed DiT architecture, dataset, training budget, and reconstruction protocol. No ICF gives the lowest dimer C_α RMSD, while Simple ICF yields the lowest dimer all-atom RMSD and the highest all-atom DockQ. This resolution-dependent result associates explicit interface context with improved atomic and in-

Table 10: Autoencoder reconstruction quality on the five standard CASP14/15 **monomer** subsets. Values report RMSD (Å, mean ± sample standard deviation, ↓) at Cα, backbone, and all-atom resolutions. Subset sizes are CASP14 T (34), T-dom (15), Oligo (39), CASP15 TS-dom (45), and Oligo (242). Baselines are CHEAP (Lu et al., 2025), ESM3 VQ-VAE (Hayes et al., 2025), ProToken (Lin et al., 2023a), DPLM-2 (Wang et al., 2025), and ProteinAE (Li et al., 2025). ProteinAE* and FoldToken-4 were re-evaluated using our test pipeline. The other baseline numbers were taken from ProteinAE (Li et al., 2025). All-atom entries for backbone-only models are marked as "−". Values are displayed to three decimal places. Shaded cells denote PLDesign. The best and second-best eligible values in each subset and resolution are shown in bold and underlined, respectively.

		CASP14			CASP15	
Resolution	**Model**	T	T-dom	Oligo	TS-dom	Oligo
Cα	CHEAP	11.150 ±9.880	8.990 ±9.180	9.930 ±10.550	10.220 ±11.230	9.220 ±12.570
	ESM3 VQ-VAE	1.020 ±1.820	0.660 ±0.420	3.080 ±7.390	1.230 ±1.260	1.940 ±2.430
	ProToken	0.990 ±0.690	0.960 ±0.580	1.150 ±1.120	1.150 ±1.000	1.180 ±0.890
	DPLM-2	1.990 ±2.030	1.870 ±1.780	2.700 ±5.050	3.310 ±5.690	3.500 ±6.230
	ProteinAE*	**0.230 ±0.150**	**0.220 ±0.110**	0.310 ±0.220	0.280 ±0.200	0.370 ±0.500
	FoldToken-4	1.967 ±1.728	5.590 ±5.176	2.330 ±2.879	2.546 ±4.394	2.790 ±4.363
	PLDesign	0.259 ±0.053	0.290 ±0.072	**0.276 ±0.074**	**0.262 ±0.087**	**0.303 ±0.118**
Backbone	CHEAP	11.160 ±10.090	4.710 ±5.210	11.100 ±11.950	10.980 ±12.440	8.240 ±14.120
	ESM3 VQ-VAE	1.280 ±2.320	0.660 ±0.420	3.110 ±7.410	1.250 ±1.280	2.470 ±2.270
	ProToken	1.140 ±0.660	1.090 ±0.160	1.550 ±1.460	1.290 ±0.670	1.330 ±0.700
	DPLM-2	1.940 ±1.990	1.470 ±0.420	3.810 ±7.200	4.580 ±6.620	3.830 ±6.900
	ProteinAE*	0.510 ±1.560	**0.230 ±0.110**	**0.310 ±0.270**	**0.310 ±0.210**	0.430 ±0.510
	FoldToken-4	1.922 ±1.730	5.557 ±5.183	2.293 ±2.874	2.507 ±4.395	2.752 ±4.367
	PLDesign	**0.353 ±0.068**	0.379 ±0.070	0.365 ±0.090	0.375 ±0.106	**0.423 ±0.207**
All atom	**PLDesign**	**0.678 ±0.547**	**0.724 ±0.378**	**0.588 ±0.075**	**0.644 ±0.273**	**0.627 ±0.158**

Table 11: Epoch-99 reconstruction across latent widths on common CASP14/15 dimer targets. ProteinAE is evaluated with the same reconstruction protocol. Values are RMSD in Å as mean ± sample standard deviation, and lower values are better. Bold and underline mark the best and second-best available values in each column.

	CASP14 dimer			CASP15 dimer		
Model	C_α	Backbone	All atom	C_α	Backbone	All atom
ProteinAE	0.963 ±0.514	1.247 ±0.529	–	0.818 ±0.317	1.122 ±0.370	–
Atom14 D8	0.484 ±0.087	0.658 ±0.091	1.160 ±0.062	0.454 ±0.104	0.625 ±0.126	1.107 ±0.067
Atom14 D16	0.505 ±0.104	0.659 ±0.113	**0.754 ±0.086**	0.489 ±0.094	0.618 ±0.142	**0.708 ±0.102**
Atom14 D32	**0.260 ±0.059**	**0.432 ±0.093**	1.026 ±0.044	**0.243 ±0.043**	**0.388 ±0.062**	0.990 ±0.046

Table 12: Post hoc D8/D16 checkpoint profile relative to the pre-specified Epoch 99 endpoint. Parentheses identify the checkpoint with the lowest observed value for each metric. Values are RMSD in Å and lower values are better. Bold and underline mark the lowest and second-lowest values within each metric and column.

Metric	Model	CASP14 minimum (epoch)	CASP14 Epoch 99	CASP15 minimum (epoch)	CASP15 Epoch 99
C_α	Atom14 D8	0.448 (73)	**0.484**	0.379 (41)	**0.454**
	Atom14 D16	**0.368 (32)**	0.505	**0.326 (32)**	0.489
Backbone	Atom14 D8	0.641 (80)	**0.658**	0.599 (80)	0.625
	Atom14 D16	**0.590 (64)**	0.659	**0.555 (64)**	**0.618**
All atom	Atom14 D8	1.160 (99)	1.160	1.107 (99)	1.107
	Atom14 D16	**0.754 (99)**	**0.754**	**0.708 (99)**	**0.708**

Table 13: Epoch-99 monomer reconstruction. Values are RMSD in Å as mean ± sample standard deviation, and lower values are better. Bold and underline mark the best and second-best values for each metric and split.

Metric	Model	CASP14 T	CASP14 T-dom	CASP14 Oligo	CASP15 TS-dom	CASP15 Oligo
C_α	ProteinAE	0.230 ±0.150	0.220 ±0.110	0.310 ±0.220	0.280 ±0.200	0.370 ±0.500
	PLDesign (Backbone)	**0.165 ±0.292**	**0.192 ±0.164**	**0.128 ±0.123**	**0.128 ±0.090**	**0.132 ±0.083**
	PLDesign (Atom14)	0.259 ±0.053	0.290 ±0.072	0.276 ±0.074	0.262 ±0.087	0.303 ±0.118
Backbone	ProteinAE	0.510 ±1.560	**0.230 ±0.110**	0.310 ±0.270	0.310 ±0.210	0.430 ±0.510
	PLDesign (Backbone)	**0.314 ±0.769**	0.362 ±0.373	**0.180 ±0.135**	**0.198 ±0.147**	**0.215 ±0.157**
	PLDesign (Atom14)	0.353 ±0.068	0.379 ±0.070	0.365 ±0.090	0.375 ±0.106	0.423 ±0.207

terchain reconstruction. It is consistent with an indirect benefit to side-chain recovery, although ICF does not specify side-chain conformations.

Table 14: ICF feature-extraction ablation on the frozen CASP14 and CASP15 dimer sets. All variants share the DiT architecture, data, training budget, decoder integration, alignment, and metric implementation. Values are mean and standard deviation across test domains. A dash marks an unavailable metric. Bold values mark the best result within each resolution.

		RMSD (Å) ↓		**DockQ ↑**	
Resolution	**Model**	CASP14 dimer	CASP15 dimer	CASP14 dimer	CASP15 dimer
C_α	No ICF	**0.470 ± 0.118**	**0.440 ± 0.097**	–	–
	Simple ICF	0.505 ± 0.104	0.489 ± 0.094	–	–
Backbone	No ICF	0.695 ± 0.155	0.635 ± 0.128	0.851 ± 0.069	0.891 ± 0.067
	Simple ICF	**0.659 ± 0.113**	**0.618 ± 0.142**	**0.865 ± 0.078**	**0.916 ± 0.060**
All atom	No ICF	1.057 ± 0.089	1.003 ± 0.075	0.864 ± 0.040	0.879 ± 0.053
	Simple ICF	**0.754 ± 0.086**	**0.708 ± 0.102**	**0.900 ± 0.029**	**0.907 ± 0.055**

Table 15: ICF feature-extraction ablation on five CASP14 and CASP15 monomer subsets. All PLDesign variants use the same architecture, data, training budget, and evaluation implementation. Values are RMSD in angstroms with mean and standard deviation. Bold values mark the best result within each resolution.

Resolution	**Model**	CASP14 T	CASP14 T-dom	CASP14 Oligo	CASP15 TS-dom	CASP15 Oligo
C_α	No ICF	**0.237 ±0.084**	**0.260 ±0.062**	**0.247 ±0.088**	**0.249 ±0.122**	**0.271 ±0.145**
	Simple ICF	0.259 ±0.053	0.290 ±0.072	0.276 ±0.074	0.262 ±0.087	0.303 ±0.118
Backbone	No ICF	0.357 ±0.142	**0.358 ±0.053**	0.383 ±0.201	**0.375 ±0.146**	**0.403 ±0.187**
	Simple ICF	**0.353 ±0.068**	0.379 ±0.070	**0.365 ±0.090**	**0.375 ±0.106**	0.423 ±0.207
All atom	No ICF	1.060 ±0.612	1.125 ±0.504	0.963 ±0.208	0.955 ±0.153	0.944 ±0.113
	Simple ICF	**0.678 ±0.547**	**0.724 ±0.378**	**0.588 ±0.075**	**0.644 ±0.273**	**0.627 ±0.158**

J METRIC-DEPENDENT ASSOCIATIONS BETWEEN GEOMETRY AND LATENT FEATURES

Figure 3 summarizes reconstruction, latent organization, and the correlation between latent displacement and interface geometry. These analyses show that the learned representation preserves structural information that can support latent trajectory scoring.

K PLDM LATENT SEARCH DETAILS

K.1 PLDM SEARCH PARAMETER EXPLORATION

The PLDM sweeps evaluate inference control while holding the trained model fixed. All numerical ablations of unconditional generation in this subsection use the PLDM EMA checkpoint, the frozen D16 autoencoder, and the same ProteinMPNN–ESMFold designability protocol. Target identities, hotspots, AF3 outputs, and binder-interface metrics do not enter hyperparameter selection. The selected configuration is transferred unchanged to target-conditional generation.

Diffusion noise scale. The SDE noise scale `sc_scale_noise` controls stochastic exploration during Euler–Maruyama integration. As shown in Table 16, increasing the scale from 0.1 to 0.2 improves designability from 50.1% to 53.2% and reduces DPT from 0.171 to 0.159. Beyond 0.2, additional noise further lowers DPT but reduces designability. At a noise scale of 0.3, the two metrics reach 0.150 and 47.8%, respectively. We select 0.2 for the subsequent search ablations, prioritizing the highest designability while retaining the diversity improvement over lower-noise settings.

Table 16: Effect of diffusion noise scale (`sc_scale_noise`) on unconditional generation. Bold and underline mark the best and second-best values in each metric.

`sc_scale_noise`	Des (↑)	DPT (↓)
0.1	50.1%	0.171
0.15	<u>51.2%</u>	0.165
0.2	**53.2%**	0.159
0.25	50.6%	<u>0.153</u>
0.3	47.8%	**0.150**

Relative latent scores. The relative scores quantify local turning-angle dispersion, local bond-length dispersion, and ensemble-referenced self-LDDT in the latent sequence. We compare these scores using the same length grid and output budget as the unconditional generation benchmark, with ten backbones at each integer length from 60 to 128 residues, totaling 690 outputs per setting (Table 18). The search configuration uses beam width $B = 4$, branching factor $K = 3$, checkpoints $\{0, 0.5, 0.75, 1\}$, FK temperature $\tau = 1.0$, step size $\Delta t = 0.0025$, and $(\eta_s, \eta_n) = (1.0, 0.2)$. For each backbone, ProteinMPNN designs eight sequences, which are evaluated through ESMFold prediction. A backbone is classified as designable when the minimum scRMSD across these predictions is at most 2 Å. DPT is computed as the mean pairwise TM-score among designable backbones within each length, followed by an equally weighted average across all 69 lengths.

Table 17: Definitions of the latent-level scores used in the PLDM inference sweeps. $\hat{u} = u/\|u\|_2$ denotes unit normalization, $\mathrm{std}(\cdot)$ the population standard deviation, and $\mathrm{CV}(\cdot) = \mathrm{std}(\cdot)/\mathrm{mean}(\cdot)$ the coefficient of variation. Every score is oriented so that higher values are better.

Scorer	Family	Formula
`lat_local_angle`	Relative	$-\mathrm{std}_{i\in\mathcal{T}}\big(\langle\hat{\delta}_{i-1,i}, \hat{\delta}_{i,i+1}\rangle\big)$
`lat_local_bond`	Relative	$-\mathrm{CV}_{i\in\mathcal{A}}\big(\|z_{i+1} - z_i\|_2\big)$
`lat_self_lddt`	Relative	$\frac{1}{\lvert\mathcal{T}_c\rvert\lvert\mathcal{Q}\rvert}\sum_{\tau\in\mathcal{T}_c}\sum_{(i,j)\in\mathcal{Q}} \mathbb{1}\big[\lvert d_{ij}(z) - \bar{d}_{ij}\rvert < \tau\big]$

Notation. $\delta_{i,j} = z_j - z_i$. For `lat_self_lddt`, $\bar{d}_{ij}$ is the batch mean of $\|z_i - z_j\|_2$ among candidates with the same residue mask at the current checkpoint, $\mathcal{Q} = \{(i,j) : i < j,\ m_i^b m_j^b = 1,\ \bar{d}_{ij} < 2\}$, and $\mathcal{T}_c = \{0.25, 0.5, 1, 2\}$.

The scores use two reference scopes. `lat_local_bond` and `lat_local_angle` evaluate the internal regularity of each candidate and remain stable under global magnitude shifts. `lat_self_lddt` compares each candidate with the current beam ensemble, using cross-candidate

agreement to suppress outlying trajectories. Their combination integrates local consistency with ensemble-level structural support.

All scores operate on the per-residue latent tensor $z \in \mathbb{R}^{L\times d}$ with $d = 16$. Binder and receptor masks $m^b, m^r \in \{0,1\}^L$ define valid residue sets $\mathcal{B} = \{i : m_i^b = 1\}$ and $\mathcal{R} = \{i : m_i^r = 1\}$. Consecutive binder pairs and triplets are $\mathcal{A} = \{i : i, i+1 \in \mathcal{B}\}$ and $\mathcal{T} = \{i : i-1, i, i+1 \in \mathcal{B}\}$. Every score is oriented so that larger values receive higher search priority.

Table 18: Unconditional generation under different latent scores. Relative features are evaluated under the same beam configuration and output budget. Bold and underline mark the best and second-best values in each metric.

Search score	**Des (↑)**	**DPT (↓)**
Default (no latent search)	53.2%	**0.159**
Relative features		
`lat_local_angle`	65.1%	<u>0.162</u>
`lat_local_bond`	61.3%	0.166
`lat_self_lddt`	**71.3%**	0.244
Mixed (angle, bond, and lDDT)	<u>69.9%</u>	0.213

All three relative latent geometry scores improve designability over the no-search baseline (Table 18), supporting their use for trajectory ranking. These gains are accompanied by higher DPT, suggesting a tendency to favor structurally similar outputs. This tendency is most pronounced for self-lDDT, which achieves the highest designability (71.3%) alongside the highest DPT (0.244). To balance designability and diversity, we adopt a combination of the angle, bond, and self-lDDT scores, with all three weights set to 1. The mixed score achieves 69.9% designability, retaining most of the gain over the no-search baseline while reducing DPT from 0.244 to 0.213 compared with self-lDDT alone.

Search checkpoints, seed pruning, and backtracking The checkpoint schedule determines when the SDE trajectory branches and latent scores are recomputed. With seed pruning and backtracking disabled, Table 19 shows comparable DPT across the three schedules (0.209–0.213). The four-point schedule $\{0, 0.5, 0.75, 1\}$ achieves the highest designability among these schedules (69.9%), exceeding the three- and five-point schedules by 4.7 and 2.6 percentage points, respectively. Compared with the five-point grid, it also requires one fewer branching and scoring step, reducing the associated computational overhead.

To further improve designability, we combine beam scoring with trajectory seed pruning and a backtracking pass (Table 20). Pruning removes low-scoring seed groups before subsequent integration, while backtracking revisits selected intermediate states and reallocates trajectories around high-scoring latent neighborhoods. Under the four-point schedule, jointly enabling seed pruning and backtracking improves designability by 4.0 percentage points, from 69.9% to 73.9%, with DPT increasing from 0.213 to 0.287.

Table 19: Effect of latent-search checkpoints on unconditional generation. Seed pruning and backtracking are enabled or disabled jointly. Bold marks the best value in each metric.

Checkpoints	**Des (↑)**	**DPT (↓)**
$\{0, 0.5, 1\}$	65.2%	**0.209**
$\{0, 0.5, 0.75, 1\}$	**69.9%**	0.213
$\{0, 0.25, 0.5, 0.75, 1\}$	67.2%	0.210

The adopted PLDM configuration uses a noise scale of 0.2, mixed relative scores with all three weights set to 1, checkpoints $\{0, 0.5, 0.75, 1\}$, a backtracking time of 0.75, and a pruning ratio of 0.5.

Table 20: Effect of the seed-pruning keep ratio under a controlled latent-search configuration. All variants use the Mixed (angle, bond, and lDDT) search score at $t \in \{0, 0.5, 0.75, 1\}$. Seed pruning is applied after scoring at $t = 0.5$, and retained high-scoring trajectories are expanded from their stored $t = 0.75$ states. Best mean values are shown in bold.

Keep ratio	**Des (↑)**	**DPT (↓)**
40%	69.0%	0.320
50%	**73.9%**	0.287
60%	72.8%	**0.265**

PD-L1 conditional seed enrichment. A matched PD-L1 ablation evaluates the PLDM checkpoint with 100 candidates at every binder length under identical ProteinMPNN, AF3, and filtering settings. Across the complete 50–120-residue range, latent trajectory search increases success from 85 of 1,500 direct samples to 114 of 1,500 candidates, a gain of 29 screen-passing binders and 34.1% relative enrichment.

Table 21: AF3-filtered PD-L1 success from the PLDM checkpoint across all evaluated binder lengths with and without latent trajectory search. Bold and underline mark the better and second-ranked sampling results.

Evaluation range	Latent search	Direct sampling	Absolute gain	Relative gain
50–120 residues	**7.60% (114 / 1,500)**	5.67% (85 / 1,500)	+29	34.1%

Table 22 localizes the aggregate gain across the complete 50–120-residue range. Nineteen of the 29 additional successes occur at lengths 50–60. Individual longer-length cells contain fewer successes and show mixed search effects, indicating that enrichment is less consistent at longer binder lengths.

Table 22: AF3-filtered successful PD-L1 binders by designed length from the PLDM checkpoint. Each sampling column contains 100 generated candidates per length. Bold and underline mark the higher and lower count within each length. Tied counts are both bold.

Binder length	Latent search	Direct sampling	Difference
50	**23**	19	+4
55	**20**	16	+4
60	**22**	11	+11
65	9	**10**	−1
70	**7**	6	+1
75	**6**	3	+3
80	6	**7**	−1
85	**6**	4	+2
90	**2**	**2**	0
95	**4**	0	+4
100	3	**4**	−1
105	**2**	1	+1
110	**3**	0	+3
115	**1**	**1**	0
120	0	**1**	−1
Total	**114**	85	+29

For a 50-residue binder, direct PLDM sampling requires 11.6 seconds per sample and latent search requires 123.8 seconds. Direct sampling supports high-throughput exploration, while latent search

concentrates computation on decoder-supported trajectories when downstream AF3 or experimental validation budgets are limited. Branch ranking remains in the D16 latent until the selected trajectories are decoded.

L BINDER BENCHMARK DETAILS

Scope of the pipeline comparison. Joint target–binder updates are evaluated as part of the PLDesign pipeline. A representation-matched estimate of their individual contribution would require a separately trained control with an explicitly factorized target code. Clamping the current target latent changes its complex-dependent input state and does not supply that control.

Full-denominator computational success. Each method has 8,000 planned candidates across six targets and all available binder lengths. Missing, unprocessed, and failed candidates remain in the denominator.

Table 23: AF3-filtered computational success over six targets and 80 target–length cells. Every method has 100 planned candidates per cell. Intervals are the cell-rate mean ± 1.96 standard errors across the 80 cells. Bold and underline mark the highest and second-highest point estimates.

Method	Successes	Rate	95% interval
PLDesign	252 / 8,000	3.15%	[2.18%, 4.12%]
BoltzGen	2,583 / 8,000	32.29%	[27.27%, 37.30%]
PXDesign	**4,876 / 8,000**	**60.95%**	[57.52%, 64.38%]
RFDiffusion3	<u>3,341 / 8,000</u>	<u>41.76%</u>	[37.94%, 45.58%]

Raw-coordinate interface geometry. Table 24 reports interface metrics computed from the generated coordinates before sequence design and AF3 prediction. For each method and metric, candidate-level values are first computed on the metric-specific valid candidate set within each target, after which target-specific summaries are averaged with equal target weight. Candidates without a raw inter-chain contact are retained only for metrics with an explicitly defined zero-contact value. They are excluded from metrics whose denominator requires a non-empty interface. Consequently, the effective denominator is metric- and method-specific rather than a single common cohort size.

PLDesign has 37 raw candidates without a heavy-atom contact pair. Contact pairs and interface quality retain all 252 candidates through defined zero-contact values. Residue-normalized raw metrics use the 215 candidates with an identifiable interface.

Raw contact count describes decoder-stage backbone geometry before sequence design and refolding. PLDesign produces fewer raw contact pairs than the coordinate-space baselines while reaching the second-highest target-side contact density (2.527), indicating a localized target-contact signal within smaller interfaces. The Atom14-supervised D16 latent retains backbone and side-chain structural context from AE training, while generated samples enter the common sequence-design and refolding pipeline as backbone seeds. The raw outputs are structurally immature complex candidates within which local interface patches can be examined. Table 3 reports interface geometry after the shared ProteinMPNN–AF3 pipeline.

L.1 AF3 CONFIDENCE AND SAMPLED TARGET VARIATION

Table 25 reports standard AF3 confidence for screen-passing candidates. For each method, we first calculate the 90th percentile within each target, then summarize the six percentiles with an equal-target mean and sample standard deviation.

The equal-target means of PLDesign's target-specific Q_{90} values are lower than those of the baselines by 3.30–3.98 pLDDT points, 0.06–0.08 in ipTM, and 0.05–0.06 in pTM. These standard scores capture global scaffold consistency and relative chain placement, complementing the local contact-density and compactness metrics in Table 3. These differences describe lower prediction confidence

Table 24: Raw interface quality across six targets using all available binder lengths. Values are target-macro means with 95% target–structure hierarchical-bootstrap intervals. Bold and underline mark the best and second-best point estimates.

Metric	Dir.	PLDesign $n = 252$	BoltzGen $n = 2{,}583$	PXDesign $n = 4{,}876$	RFDiffusion3 $n = 3{,}341$
Contact pairs	↑	23.330 [16.221, 29.807]	**64.928 [58.479, 72.113]**	31.032 [26.885, 35.577]	<u>47.392 [41.231, 53.859]</u>
Contacts / target-interface residue	↑	<u>2.527 [2.069, 2.911]</u>	**2.662 [2.611, 2.729]**	1.745 [1.683, 1.813]	2.410 [2.336, 2.486]
Contacts / binder-interface residue	↑	2.195 [1.951, 2.444]	**2.751 [2.709, 2.805]**	1.870 [1.827, 1.915]	<u>2.358 [2.330, 2.385]</u>
Contacts / total interface residue	↑	1.137 [0.987, 1.270]	**1.345 [1.328, 1.364]**	0.895 [0.873, 0.916]	<u>1.183 [1.165, 1.201]</u>
Compact local score	↑	1.129 [0.922, 1.319]	**1.594 [1.539, 1.654]**	0.754 [0.715, 0.796]	<u>1.233 [1.223, 1.244]</u>
Interface quality score	↑	0.075 [0.050, 0.098]	**0.185 [0.175, 0.195]**	0.069 [0.067, 0.072]	<u>0.123 [0.113, 0.133]</u>

Table 25: Target-macro AF3 confidence among screen-passing candidates. Values are equal-target means and sample standard deviations of target-specific 90th percentiles. Bold and underline mark the highest and second-highest values.

Method	pLDDT Q_{90}	ipTM Q_{90}	pTM Q_{90}
PLDesign	89.22 ± 1.48	0.82 ± 0.03	0.86 ± 0.03
PXDesign	<u>92.88 ± 1.66</u>	$\mathbf{0.90 \pm 0.02}$	$\mathbf{0.92 \pm 0.02}$
BoltzGen	92.52 ± 1.51	0.88 ± 0.04	<u>0.91 ± 0.02</u>
RFDiffusion3	$\mathbf{93.20 \pm 1.53}$	<u>0.89 ± 0.01</u>	<u>0.91 ± 0.01</u>

within the retained PLDesign cohort. AF3 uses the reference target as its template, whereas PLDesign permits target variation during sampling. The present comparison does not isolate the effect of this difference on confidence.

Table 26 jointly measures local target variation and preservation of target-side interface participation after the common AF3 prediction protocol. PLDesign samples the broadest local target distribution around the hotspot. Among the 215 candidates with a raw heavy-atom interface, the equal-target macro-averaged proportions retaining at least 80% and all target-interface residues are 17.5% and 12.2%, respectively.

Residue retention tracks target-side interface participation under sampled target variation and is interpreted alongside the AF3-refolded contact-efficiency results.

L.2 Binder structural retrieval across target contexts

We paired binder–training-chain retrieval with design-target–partner comparison to characterize the target contexts associated with retrieved structures. Each generated binder was matched to the training-dimer chain with the highest Foldseek alignment TM. Candidate partners were drawn from the extracted partner-chain set and identified as other chains in the same PDB entry and structural model as the retrieved chain. This co-assembly rule provides a proxy for annotated interface partners. For each binder, we retained the maximum strict TM, $\min(\mathrm{qTM}, \mathrm{tTM})$, between its design target and the candidate partners with available comparisons.

Table 26: Target motion and raw target-interface residue retention across six targets and all available binder lengths. Motion is measured after global target alignment. Retention is measured after reference-target-templated AF3 prediction. Counts are totals, and other values are equal-target macro means or proportions. Motion statistics describe sampled variation without assigning a preferred direction.

Metric	PLDesign	BoltzGen	PXDesign	RFDiffusion3
Motion n	252	2,583	4,876	3,341
10 Å backbone RMSD	0.708	0.115	0.243	0.029
Motion ≥ 0.5 Å	90.6%	0.0%	17.8%	0.0%
Motion ≥ 1 Å	6.1%	0.0%	0.0%	0.0%
Fixed-frame TM	0.502	0.950	0.966	0.997
5 Å C_α RMSD	0.555	0.094	0.083	0.024
Retention n	215	2,583	4,876	3,341
Retention $\geq 80\%$	17.5%	71.5%	11.4%	84.6%
Retention $= 100\%$	12.2%	7.5%	0.5%	21.4%

Of 252 binders, 242 have both comparisons available. Eight have no retrieved training chain, and two lack a target–partner comparison. The mean maximum target–partner strict TM is 0.222 ± 0.105, with 239 of 242 candidates (98.8%) below 0.5. Among the 209 candidates with binder–chain alignment TM ≥ 0.7, 207 (99.0%) fall below this threshold, with a mean maximum strict TM of 0.224 ± 0.104. The three maxima above 0.5 in the complete comparable set are 0.961 and 0.752 for BHRF1 and 0.661 for TrkA.

The nearest-chain analysis characterizes structural similarity within the retrieved chain and candidate-partner sets. Binder retrieval uses alignment TM, and target–partner comparison uses chain-length-normalized strict TM. Interface geometry and local target motifs require separate comparisons.

Table 27: Binder structural retrieval and associated target–partner similarity. Each generated binder is matched to the training-dimer chain with the highest Foldseek alignment TM. Candidate partners are other chains in the same PDB entry and structural model within the extracted partner-chain set, providing a co-assembly proxy for annotated interface partners. For each binder, target–partner similarity is the maximum strict TM, $\min(\mathrm{qTM}, \mathrm{tTM})$, over candidate partners with available comparisons. Of 252 binders, 8 have no retrieved training chain and 2 lack a target–partner comparison, leaving $N = 242$ comparable candidates. Continuous values are mean $\pm$ sample standard deviation. Percentages use the row-specific N, except in the final column, which reports the fraction among candidates with binder–chain alignment TM ≥ 0.7.

		Binder vs. training chain	Design target vs. candidate partners		Conditional comparison
Design target	N	Alignment TM ≥ 0.7	Maximum strict TM	Maximum strict TM < 0.5	Maximum strict TM < 0.5 among alignment TM ≥ 0.7
BHRF1	22	20 (90.9%)	0.285 ± 0.210	20 (90.9%)	18/20 (90.0%)
InsulinR	37	33 (89.2%)	0.204 ± 0.063	37 (100.0%)	33/33 (100.0%)
PD-L1	110	94 (85.5%)	0.223 ± 0.083	110 (100.0%)	94/94 (100.0%)
IL-7RA	18	16 (88.9%)	0.181 ± 0.101	18 (100.0%)	16/16 (100.0%)
TrkA	49	42 (85.7%)	0.224 ± 0.104	48 (98.0%)	42/42 (100.0%)
SC2RBD	6	4 (66.7%)	0.202 ± 0.054	6 (100.0%)	4/4 (100.0%)
Overall	242	209 (86.4%)	0.222 ± 0.105	239 (98.8%)	**207/209 (99.0%)**

Table 28: Inference time and peak GPU memory usage of different methods. Lower values are better. Bold and underline mark the best and second-best values in each column.

Method	Inference time per sample (s)	Peak GPU memory (GiB)
BoltzGen	17.4	4.3
PXDesign	**11.4**	**1.4**
RFDiffusion3	17.2	3.9
Proteina-Complexa	431.3	26.7
PLDesign (direct)	<u>11.6</u>	3.8
PLDesign (latent search)	123.8	<u>3.1</u>

M Inference Efficiency Evaluation Details

We evaluated the inference efficiency of PLDesign against baseline models by generating 20 binders of length 50 for the PD-L1 target. All evaluations were conducted using a batch size of four on a single NVIDIA V100 GPU with 32 GB of memory. Proteina-Complexa and PLDesign (latent search) both used a beam width of $B = 4$. The runtime is normalized by the sample count to report the average seconds per sample, and the memory usage reflects the recorded GPU peak memory during the generation process.

N Broader Impacts

The development of computational methods for protein binder design, such as PLDesign, carries some broader impacts. On the positive side, these tools can accelerate the design of therapeutic binders and antibodies, reducing the time and wet-lab costs associated with traditional discovery pipelines. This acceleration has the potential to facilitate rapid responses to emerging diseases and improve the accessibility of novel therapeutics.