DESIGNING PROTEIN STRUCTURES FROM NATURAL LANGUAGE

Anonymous authors
Paper under double-blind review

ABSTRACT

A central challenge in *de novo* protein design is generating proteins with desired structural and functional properties. Existing design models typically require a dedicated input representation for each property, limiting the design objectives they support. We develop **Prompteina**, the first all-atom structure generation model that supports new and broader design objectives directly specified in natural language, analogous to text-to-image generation. To train Prompteina, we construct **Prompteina-Data**, a paired text–protein structure dataset of 1.8M structures and 50M+ captions describing global and residue-specific structural and functional properties. We then adapt Proteina-Complexa, a pretrained structure generation model, through text-conditioned pretraining and finetuning that aligns residues with their corresponding text. Prompteina outperforms evaluated baselines on 15 of 18 structural properties and generates proteins that more closely match descriptions of their intended functions. Prompteina outperforms state-of-the-art binder-design models on at least 13 of 20 targets by up to 14%, while supporting specifications beyond existing conditioning interfaces. With iterative in silico feedback, Prompteina's gain over non-iterative generation is 57% larger than the next-best baseline on the hardest targets. Finally, we experimentally test Prompteina designs against GM2A, which has no reported de novo binders, and identify one binder whose binding signal increases with concentration and shows both association and dissociation. These results establish natural language as an effective and powerful conditioning interface for protein structure generation.

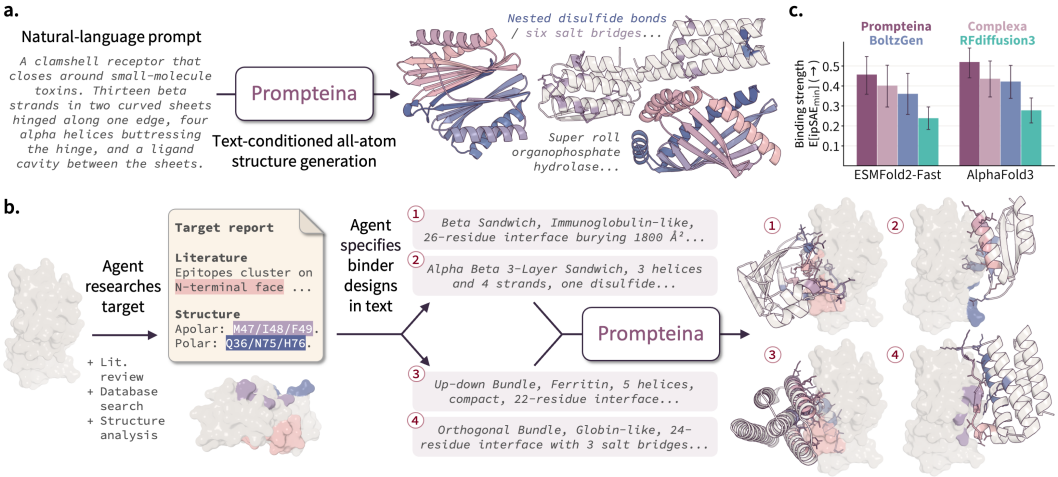


Figure 1: **Prompteina generates all-atom protein structures directly from natural language.** **(a)** A single natural-language interface supports diverse design objectives spanning global and residue-specific properties. For full prompts, see App. E.4. **(b)** In agentic binder design, an agent researches target biology and writes detailed, research-informed natural-language prompts for candidate binders. Given these prompts and the target structure, Prompteina's expressive natural-language interface allows these specifications to be realized in generated structures. **(c)** Across 20 targets, Prompteina achieves the highest predicted binding scores with multiple structure predictors.

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.

1 INTRODUCTION

A central challenge in *de novo* protein design is generating proteins with specified properties, from their overall fold or function to properties of the individual residues that make up the protein. Existing protein structure generation models allow control over such properties, but a custom input representation must typically be engineered for each property to condition generation (Watson et al., 2023; Geffner et al., 2025; Stark et al., 2026). For instance, a protein's desired fold may be represented by a single categorical label, and its desired secondary structure by a sequence that labels each residue as part of a helix, strand, or loop. While this form of conditioning provides precise control, the need to define a separate representation for each property limits the range of design objectives that any one model can support.

An ideal conditioning approach would support a diverse set of properties without the need to engineer a custom representation for each. Image generation provides a precedent: natural language offers a unified conditioning interface for expressing diverse visual attributes that would otherwise require task-specific labels or controls (Reed et al., 2016; Rombach et al., 2022).

We introduce **Prompteina**, the first text-conditioned all-atom protein structure generation model that supports diverse design objectives through a single natural-language interface. Given an input prompt, Prompteina jointly generates an all-atom protein structure and sequence conditioned on the prompt's text embeddings. By contrast, existing approaches either use text to guide structure generation only at inference time (Ingraham et al., 2023) or use it to generate sequences that must then be folded into structures (Dai et al., 2025; Pei et al., 2026). This limits robust, fine-grained control over both structure and sequence. By directly conditioning the generation of both modalities on text, Prompteina enables steering of global properties such as fold class alongside residue-specific properties including specific disulfide bonds and binder–target contacts (Fig. 1). Prompteina also complements recent work on agentic protein design by letting language-model agents pass specifications directly to the structure generator in natural language rather than translate them into fixed, model-specific inputs that constrain which design decisions they can express (Shanehsazzadeh, 2026).

Training Prompteina for direct, fine-grained control requires paired text–structure data describing both global and residue-specific properties. Existing text–protein datasets, however, pair functional descriptions with sequences and lack descriptions of structural features (Dai et al., 2025; Song et al., 2026). We therefore construct **Prompteina-Data**, comprising approximately 1.8M structures with over 50M captions describing both functional and structural properties. We transform annotations from databases and scientific literature into diverse natural-language captions and annotate, where possible, correspondences between phrases in the captions and the residues they describe (§2.1). Using this data, we extend Complexa (Didi et al., 2026), a pretrained all-atom protein structure generation model, with two text-conditioning pathways (§2.2). We adapt the model at scale through text-conditioned continual pretraining on Prompteina-Data, then fine-tune it with direct supervision of the correspondences between caption text and residues for finer residue-level control (§2.3).

Across five groups of structural properties, Prompteina leads evaluated baselines on 15 of 18 tasks, supports more properties than any single baseline, and generates proteins that more closely match descriptions of their intended functions (§§3.1.1–3.1.2). At finer structural scales, Prompteina steers the number of residues at the binder–target interface across 20 targets ($\rho > 0.99$, §3.2.1). Beyond steering individual interface properties, we evaluate Prompteina in agent-guided protein binder design. The same language-model agent writes specifications based on the properties supported by each model's conditioning interface. Prompteina outperforms every baseline on at least 13 of 20 targets and raises predicted binding scores by 14% over the strongest baseline (§3.2.2). When we allow the agent to iteratively refine designs, Prompteina shows larger gains over matched-budget i.i.d. generation than either baseline, especially on the most difficult targets (§3.2.3). As a proof of concept, we conducted a prospective, agent-guided design campaign against GM2 ganglioside activator protein (GM2A), a therapeutic target implicated in GM2 gangliosidosis (Deschenes et al., 2023). GM2A has no reported *de novo* binders and was one of three targets without a hit in a prior campaign (Stark et al., 2026). In our campaign, one of 28 expressed designs showed a concentration-dependent signal in a binding assay, suggesting a possible GM2A binder for further testing (§3.3).

These results establish natural language as an effective conditioning interface, enabling structural and functional control across scales and integrating language-based reasoning and iterative refinement into protein design. We will release our data, code, and model weights to support future work.

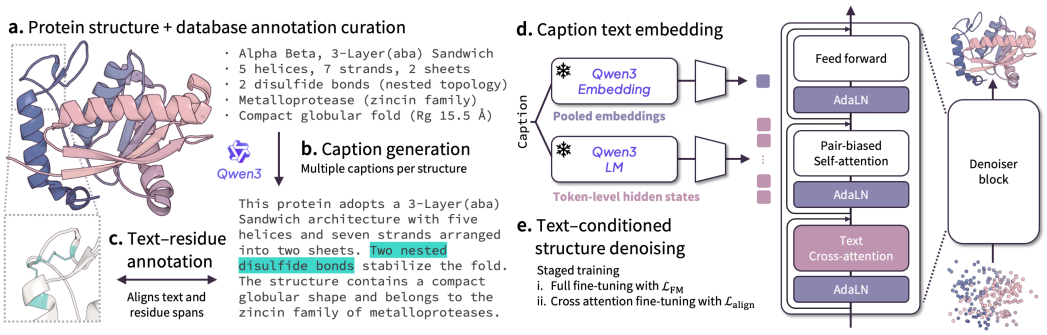


Figure 2: **Paired text–structure data and text-conditioned structure generation**. **(a)** Structural and functional annotations are curated and **(b)** passed to a large language model for caption generation. **(c)** Residues are aligned to their corresponding caption text spans for supervised alignment fine-tuning. **(d)** Captions are embedded into pooled and per-token text representations, then projected to condition the structure generation model. **(e)** The denoiser is first trained on caption–structure pairs with a flow-matching objective, then fine-tuned with supervised text–residue alignment (§2.2).

2 Text-Guided Protein Structure Generation

A natural-language conditioning interface should let protein designers specify properties across structural scales, from global properties of the whole protein, such as fold class and overall shape, to residue-specific properties, such as disulfide bonds, catalytic sites, and binder–target contacts. We therefore construct Prompteina-Data, a dataset pairing protein structures with natural-language descriptions spanning these scales (§2.1), and use it to train Prompteina, an all-atom structure generation model conditioned on natural language for monomer and binder design (Fig. 2; §§2.2–2.3).

2.1 Prompteina-Data: Text–Structure Data

We construct Prompteina-Data from experimental structures in the Protein Data Bank (PDB; Berman et al., 2000), predicted structures in the AlphaFold Database (AFDB; Varadi et al., 2022), and Teddymer synthetic dimers (Didi et al., 2026) (App. B.1). Prompteina generates one chain at a time, so we treat each chain as a separate entry, giving about 1.8M entries. The dataset includes up to 59 annotations per entry, derived from database fields and atomic coordinates (Fig. 2a; App. B.2). Annotations span global properties, including fold class and compactness, and residue-specific properties with residue positions, such as disulfide bonds, catalytic sites, and protein–protein contacts.

At inference, prompts may specify one property or combine several, so we vary caption content and length in the training data (Wu et al., 2025) (App. B.3). For each entry, we sample several subsets that differ in the number and choice of annotations, then prompt Qwen3-235B-A22B (Yang et al., 2025) to write one caption from each subset (Fig. 2b; App. B.4). In total, we generate 52.4M captions, averaging 29 captions per entry. We use an LLM judge to compare 80 generated captions with their ground-truth annotations (App. B.5). Across this sample, the generated captions match their ground-truth annotations closely (97.8% recall, 89.8% precision, and 95.7% accuracy).

2.2 Prompteina: Text-Conditioned Structure Generation

Our model, Prompteina, builds on the pretrained Complexa model (Didi et al., 2026) (App. C.1), adding two text-conditioning pathways and an auxiliary text–residue alignment objective.

Global and residue-level text conditioning. Complexa's denoiser generates backbone coordinates and one side-chain latent per residue before a frozen variational autoencoder decodes both into an all-atom structure and amino acid sequence. To give every residue the same conditioning signal for global properties, we broadcast a pooled embedding from frozen Qwen3-Embedding-8B (Zhang et al., 2025) through adaptive layer normalization. For residue-specific properties, each residue attends to frozen Qwen3-8B token states (Yang et al., 2025) via cross-attention before protein self-attention in each denoiser layer (Fig. 2d; Apps. C.2–C.3). Two-layer MLPs map both

representations to denoiser dimensions. We ablate pooled conditioning, cross-attention, and their combination, which performs best across tasks (App. D).

Text–residue alignment. In text-conditioned pretraining, we train the denoiser using a flow-matching loss but do not directly supervise which residues correspond to each text span. Prompteina-Data contains residue positions for many annotations, which we use to construct text–residue alignment annotations (Fig. 2c). For each Prompteina-Data annotation selected for a caption, we call its associated residue positions the annotation's residue span. A residue span contains, for example, the helix residues for a mainly alpha fold annotation and the participating residues for a disulfide bond or protein–protein contact. We attempt to locate the corresponding text span with regular expressions and TF-IDF-weighted token matching. If either span is unavailable, we retain the caption–structure pair but omit that alignment annotation. We validate span matching on held-out captions (App. B.6).

We train an auxiliary alignment head on these annotations, following work using segmentation masks for token-level supervision in text-to-image diffusion (Wang et al., 2024). For N protein residues and T caption tokens, we represent alignment annotations as a binary matrix $\mathbf{M} \in \{0,1\}^{N\times T}$, where $M_{n,i} = 1$ when token i is aligned to residue n. The alignment head predicts a logit $A_{n,i}$ using cross-attention updates for every residue–token pair (App. C.4). Let $\mathcal{V} \subseteq \{1,\dots,N\}\times\{1,\dots,T\}$ contain residue–token pairs used for alignment supervision, and let σ denote the sigmoid function. Aligned residue–token pairs are sparse, so for each supervised example we set w_+ to the ratio of negative to positive pairs (App. C.4). The alignment loss is

$$\mathcal{L}_{\text{align}} = -\frac{1}{|\mathcal{V}|}\sum_{(n,i)\in\mathcal{V}} \left[w_+ M_{n,i}\log\sigma(A_{n,i}) + (1-M_{n,i})\log(1-\sigma(A_{n,i}))\right]. \tag{1}$$

After alignment fine-tuning, Prompteina gains finer control over features such as disulfide-bond and salt-bridge counts and retains global fold control without the alignment head at inference (Table 9).

2.3 Training and Inference

Training. We first adapt the Complexa denoiser to natural-language conditioning using newly initialized parameters for pooled and token-level text conditioning (§2.2; App. C.5). These parameters are first trained with the denoiser frozen before we unfreeze it for joint text-conditioned continual pretraining on Prompteina-Data, following staged training in text-to-image models (Chen et al., 2023) (Fig. 2e; App. C.6). For alignment fine-tuning, we freeze the denoiser and update text projectors, cross-attention layers, and alignment head. We describe other Complexa training modifications that improve performance and their ablations (Apps. C.6, D).

Inference. At inference, Prompteina takes a prompt and protein length and, for binder design, a target structure and optional target contact residues. Prompteina samples backbone coordinates and side-chain latents with the same stochastic differential equation as Complexa and uses classifier-free guidance to adjust text-conditioning strength (Ho & Salimans, 2022) (App. C.7). The frozen variational autoencoder decodes these into an all-atom structure and amino acid sequence.

3 Experiments

We show that Prompteina's natural-language conditioning interface controls structure generation from prompts specifying structural and functional properties (§3.1). This control extends to protein binding, where Prompteina directly steers the binder–target interface (§3.2.1) and converts agent-written design specifications into stronger predicted binders than state-of-the-art binder design models (§3.2.2). Prompteina's natural-language conditioning interface also improves predicted binding most when specifications are revised iteratively, especially on the hardest targets (§3.2.3). Finally, we prospectively design and experimentally test candidates against GM2A, identifying one with a concentration-dependent binding signal (§3.3).

3.1 Text-Guided Structure Generation

We first test Prompteina on prompts for structural properties (§3.1.1) or protein function (§3.1.2).

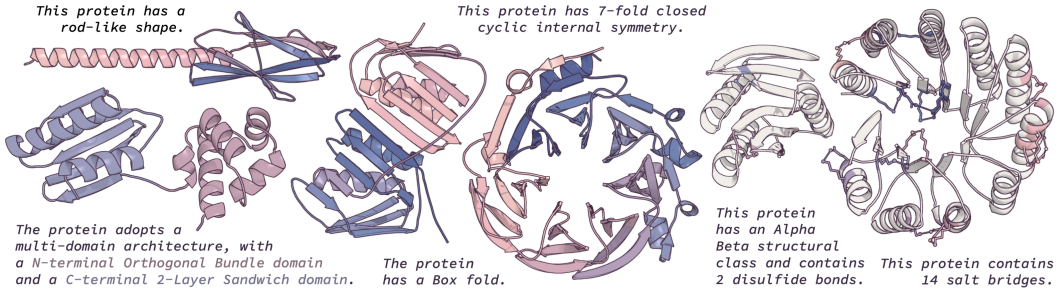


Figure 3: **Natural language controls global and residue-specific structural properties.** Examples span global properties (shape, architecture, domain composition, and symmetry), residue-specific properties (disulfide and salt-bridge counts), and combinations of both (fold class with disulfide count). All structures match their captions and are designable (scRMSD < 2 Å).

Table 1: **Prompteina controls the broadest range of structural properties. (a) Text-guided models.** All models receive identical free-form prompts for 18 tasks in five property columns. Cells average task scores by column; parentheses give task counts (full results in Table 10). † denotes accuracy within 2.1 pp of chance (33.3%). **(b) Property-specific models.** Baselines use property-specific labels or structured inputs. Columns show tasks supported by at least one baseline; dashes mark unsupported tasks. We report within-prompt half-widths of 95% stratified-bootstrap intervals.

(a)	Accuracy (%) (↑)		Spearman ρ (↑)		Accuracy (%) / ρ (↑)
Method	Fold (6)	Shape (3)	Secondary (3)	Tertiary (2)	Fold + sec./tert. (4)
Prompteina	**53.1**	**62.2**	**0.82**	**0.48**	**90.2 / 0.60**
Chroma (text)	10.9	34.2†	0.52	0.35	33.9† / 0.33
Pinal	12.7	35.1†	0.38	0.17	35.4† / 0.12
BioMatrix	9.9	32.7†	0.32	0.18	31.4† / 0.15

(b)	Accuracy (%) (↑)			Spearman ρ (↑)		
Method	CATH class	CATH arch.	Compactness	Helix	Strand	Sheet
Prompteina	96.0 $\pm$ 3.0	**51.1** $\pm$ 1.6	**76.7** $\pm$ 5.3	**0.94** $\pm$ 0.01	0.88 $\pm$ 0.02	0.64 $\pm$ 0.04
Chroma (label)	64.7 $\pm$ 7.7	7.5 $\pm$ 1.0	39.3 $\pm$ 5.3	0.86 $\pm$ 0.02	0.58 $\pm$ 0.04	0.59 $\pm$ 0.05
Proteina	**96.7** $\pm$ 3.0	27.5 $\pm$ 1.4	—	—	—	—
RFdiffusion	—	—	—	0.87 $\pm$ 0.01	**0.92** $\pm$ 0.01	**0.82** $\pm$ 0.02

3.1.1 Control of Structural Properties

Baselines. For structural control, we benchmark text-guided models that accept free-form text prompts and property-specific models that take labels or structured inputs for supported properties. Text-guided models, comprising Chroma (text), Pinal, and BioMatrix, receive the same free-form prompts as Prompteina; property-specific models, comprising Chroma (label), Proteina, and RFdiffusion, receive their native labels or structured inputs. For each baseline, we specify every property it supports and evaluate it only on tasks expressible through its conditioning interface (App. E).

Evaluation. We compare Prompteina with six baselines across 18 tasks organized into five property groups: fold and topology, secondary structure, tertiary structure, global shape and surface properties, and combinations of fold class with secondary-structure or disulfide counts (Table 1a). Complete task definitions and scoring procedures are included in App. E. For categorical properties, we report accuracy, and for count-valued properties such as helix count, we report the Spearman rank correlation (ρ) between requested and realized values measured in each generated structure.

Prompteina enables new and broader forms of structural control. Across the benchmark, Prompteina controls all five property groups and leads on 15 of 18 tasks (Table 1a and Table 10). It also enables CATH labels for multiple domains in one protein and prompts specifying multiple structural constraints, such as fold class and helix or disulfide count (Table 10). With one natural-language interface, Prompteina nearly matches Proteina on CATH class and leads property-specific

models on CATH architecture, compactness, and helix count (Table 1b). RFdiffusion performs better on strand and sheet counts because it accepts per-residue secondary-structure inputs. Prompteina's structural control is robust to paraphrasing (App. E.1) and preserves designability and novelty, with a higher designable fraction than every baseline on shared tasks and more novel designable structures than Complexa (Tables 13 and 14). We show representative Prompteina samples in Fig. 3, with additional designable examples and their refolds in App. Fig. 6.

3.1.2 Generation from Functional Specifications

We next test whether Prompteina can generate structures for protein families, such as lysozymes and green fluorescent proteins, from functional prompts. Functional prompts lack a directly verifiable structural property, so we evaluate each protein with ProTrek (Su et al., 2025), a contrastive encoder of sequences, structures, and text. We report ProTrek similarity between each prompt and its generated sequence (Seq. sim.) or structure (Struct. sim.). A high ProTrek score is less meaningful for a low-confidence predicted fold, so we also weight each score by predicted-structure confidence (pLDDT). Each model receives four prompt formats across 122 families, including at least one format drawn from its training distribution.

Table 2: **Prompteina leads on functional prompts.** Each cell reports a model's highest mean across four prompt formats, with a 95% bootstrap CI over 122 families. pLDDT-weighted scores multiply each ProTrek similarity by mean per-residue pLDDT.

	Seq. sim. (↑)		**Struct. sim.** (↑)	
Method	Unweighted	pLDDT-wtd.	Unweighted	pLDDT-wtd.
Prompteina	**8.46** ± 0.98	**7.12** ± 0.88	**7.36** ± 1.04	**6.20** ± 0.90
Pinal	7.55 ± 0.79	5.98 ± 0.70	7.09 ± 0.73	5.49 ± 0.68
BioMatrix	5.73 ± 0.68	3.70 ± 0.58	4.64 ± 0.63	3.20 ± 0.56

Prompteina generates structures consistent with functional specifications. At each model's best prompt format, Prompteina achieves the highest mean on both ProTrek similarity metrics (Table 2). With pLDDT weighting, it significantly outperforms BioMatrix on both metrics and Pinal on sequence similarity, while retaining the highest mean ProTrek structure similarity (Table 2; App. G).

3.2 Text-Guided Protein Binder Design

Designing strong binders requires control over both global binder properties, such as fold class, and the size and chemistry of the binder–target interface (Cao et al., 2022; Watson et al., 2023). We test whether Prompteina can steer these properties through natural language (§3.2.1) and produce better predicted binders than methods using predefined, property-specific inputs (§§3.2.2–3.2.3).

3.2.1 Binder–Target Interface Steering

We first test control over binder–target interface size, which is associated with binding specificity (Prasad Bahadur et al., 2004). We use interface residue count, defined as the number of binder residues that contact the target, as a proxy (App. H.1). Across 20 targets from the Complexa binder-design benchmark (Didi et al., 2026), we generate binders with Prompteina while requesting interface residue counts from 8 to 30 and holding the target and binder-length distribution fixed. For each target, we compare requested and realized counts using Spearman correlation (ρ) and a linear fit.

Prompteina reliably steers binder–target interface residue count. The mean realized count tracks the requested count closely and nearly linearly across all 20 targets (Spearman's $\rho = 0.994$–1.000 and linear-fit $R^2 = 0.916$–0.996; App. Fig. 7). Across the 20 targets, Prompteina steers interface residue counts below and above unconditional generation, allowing the size of the resulting binder–target interface to be tuned (App. Fig. 7). Figure 4 shows binders for the same target with low and high realized counts. This control is robust to prompt paraphrasing (Table 18).

3.2.2 Agentic Protein Binder Design

Designing a binder against a new target begins with research into the shape and chemistry of its surface, which epitopes known partners engage, and what scaffolds have succeeded against related targets. Previous work shows that LLM agents can conduct this research and operate design tools end to end (Ghafarollahi & Buehler, 2024; Shanehsazzadeh, 2026). However, an agent can communicate

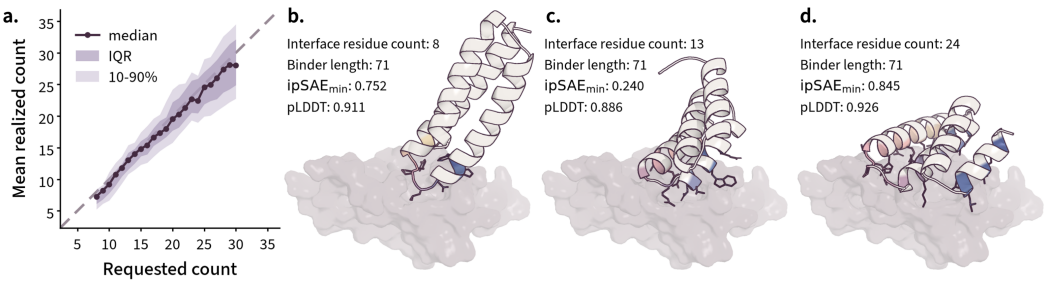


Figure 4: **Prompteina steers binder–target interface residue count. (a)** Requested versus mean realized count across 20 targets. The line shows the median across targets, the bands show the interquartile and 10th–90th percentile ranges, and the dashed line marks equality. **(b–d)** Predicted TrkA binder–target complexes whose realized counts are within one residue of the requested counts (requested: 8, 14, 24; realized: 8, 13, 24). Binders (white cartoons) contact the target (surface) through interface residues (colored sticks). For each complex, we report ipSAE$_{\min}$ (↑) as a proxy for binding strength and complex-averaged pLDDT (↑) as a proxy for structural confidence.

design decisions only through the conditioning inputs a model accepts. Existing models require a separate input representation for each property and therefore support a narrower range of properties, whereas Prompteina's single natural-language interface supports a broader range. We test whether this broader range of supported properties allows the same agent to produce better binders.

Setup and baselines. We compare Prompteina with Complexa (Didi et al., 2026), RFdiffusion3 (Butcher et al., 2025), and BoltzGen (Stark et al., 2026) on Complexa's 20-target benchmark. For each target, Claude Opus 4.6 compiles a research dossier and writes 200 binder specifications using each model's supported input properties (App. H.2). Each model samples one structure per specification. We also run all three baselines in a non-agentic setting, where they receive only the target structure, binder-length range, and fixed benchmark hotspots. We do not run Complexa with the agent because its released checkpoint supports no additional input properties.

Evaluation. We score each design under three sequence conditions: the generated sequence (`self`), full ProteinMPNN redesign (`MPNN`), and fixed-interface redesign (`MPNN-FI`) (Dauparas et al., 2022). We find that ipSAE$_{\min}$ from ESMFold2-Fast separates binders from non-binders significantly better than the AlphaProteo AF2 filters used by Complexa (App. H.4), corroborating Shanehsazzadeh (2026), and hence use it as a proxy for predicted binding strength.

To compare methods across design budgets, let $M(k)$ be the mean ipSAE$_{\min}$ of a target's k best designs, out of N generated. We report the expected top-k ipSAE$_{\min}$,

$$\mathbb{E}\big[M(K)\big] = \Big(\textstyle\sum_{k=1}^{N} \frac{1}{k}\Big)^{-1} \sum_{k=1}^{N} M(k)/k. \tag{2}$$

This quantity averages $M(k)$ under a log-uniform prior on K, weighting equally each order of magnitude of k. We compute the statistic separately for each target before averaging across all 20, and report the full comparison across design budgets for each sequence condition in App. H.6.

Prompteina produces the strongest predicted binders. Prompteina achieves the highest expected top-k ipSAE$_{\min}$ in all three sequence conditions (Table 3). Prompteina's lead is largest for the generated sequence and fixed-interface redesign, narrows under full ProteinMPNN redesign, and is robust to AlphaFold3 evaluation of the generated sequence (Abramson et al., 2024) (Table 22). RFdiffusion3 and BoltzGen achieve similar predicted binding scores with or without the agent, and all four variants score below Prompteina (Table 3). Under redesign, Prompteina's top designs are more diverse than either BoltzGen variant and match agent-guided RFdiffusion3 (App. H.9).

Predicted binding improves with interface properties. Prompteina lets the agent specify binder–target interface size through residue count and buried surface area, neither of which RFdiffusion3 or BoltzGen accepts (App. A). Prompteina generates larger interfaces than either baseline, and within each target greater realized buried surface area is associated with higher predicted binding scores (App. H.7). To test this directly, we rerun Prompteina on all 20 targets, first specifying

overall fold, then adding shape, and finally adding interface properties. We find only a modest gain from adding shape, but the largest gain from interface properties, suggesting that direct specification of the latter is especially important (Table 26).

Prompteina's language interface preserves rich design intent for generation. For CD45, we trace how each model represents agent-selected binder–target interface properties (Fig. 1b). Claude identifies distinct CD45 sites, including the charged and polar BC8 epitope and hydrophobic M47/I48/F49 patch, then proposes a fold and interface geometry for each. All three models receive the same CD45 structure and research dossier, but only Prompteina can condition on the resulting fold, interface geometry, size, and chemistry in natural language. For RFdiffusion3, Claude encodes a non-loop flag and, in some designs, atom-level hydrogen-bond or burial labels. For BoltzGen, Claude encodes residue-level helix, sheet, and loop ranges (App. H.2.5). Using these richer specifications, Prompteina generates stronger predicted binders: its two best designs outscore every baseline design while targeting different CD45 sites and adopting beta-rich and alpha-helical folds.

Table 3: **Prompteina produces the strongest predicted binders.** We evaluate 200 designs per method on 20 targets. Expected top-k ipSAE$_{\min}$ (Eq. 2) measures predicted binding in each sequence condition. Diversity is one minus the mean pairwise TM-score among each method's 32 best `MPNN-FI` designs on 18 targets with at least 32 positive-scoring designs per method.

		Top-k ipSAE$_{\min}$ (↑)			**Diversity (↑)**
Method	**Agent**	self	MPNN	MPNN-FI	1−TM sim.
Prompteina	✓	**0.455**	**0.584**	**0.549**	<u>0.584</u>
Complexa	✗	<u>0.400</u>	0.538	<u>0.510</u>	0.520
BoltzGen	✓	0.359	0.575	0.490	0.516
BoltzGen	✗	0.376	<u>0.577</u>	0.493	0.399
RFdiffusion3	✓	0.237	0.559	0.454	**0.587**
RFdiffusion3	✗	0.251	0.563	0.469	0.541

3.2.3 Iterative Binder Prompt Optimization

Recent design models perform inference-time search by scoring partial structures during generation and retaining promising candidates (Didi et al., 2026; Rector-Brooks et al., 2026). Analogously, when agents call design models as tools, they can use in-silico or experimental feedback on completed designs to identify promising design strategies and revise the next batch of specifications (Shanehsazzadeh, 2026). We test whether this iterative revision yields larger gains when the agent uses Prompteina's broader conditioning interface than when it uses RFdiffusion3 or BoltzGen. For each method and target, we run four rounds of 50 designs, matching the 200-design budget of §3.2.2. Round 1 contains 50 i.i.d. designs generated as in §3.2.2. Before each later round, the agent reads all scores so far and writes 50 new specifications (App. H.10). At budgets of 50, 100, 150, and 200 designs, we compare the accumulated iterative pool with matched-budget i.i.d. generation.

Prompteina gains most from iterative optimization on the hardest targets. Across 20 targets, iterative feedback benefits Prompteina more than either baseline. The advantage is largest on the hardest targets: stratifying by difficulty (App. H.11), Prompteina shows larger top-1 and top-8 gains than either baseline (Table 31). Using Claudin-1 as an example, we find that the agent first tests new topologies, then requests a larger interface, improving the best design after each round (Fig. 5; App. H.10). Paired with either baseline, the agent also identifies productive patterns, but each model's interface offers fewer properties to revise (App. H.10). This suggests that Prompteina's broader conditioning lets the agent make more useful changes in response to in-silico feedback.

3.3 Prospective GM2A Binder Campaign

As a proof of concept, we conducted a prospective, agent-guided design campaign against GM2 ganglioside activator protein (GM2A), a therapeutic target implicated in GM2 gangliosidosis (Deschenes et al., 2023). We chose GM2A because it has no reported *de novo* binders and was one of three targets with no hit in a prior campaign (Stark et al., 2026). We ran four parallel iterative-optimization threads and selected 90 designs for testing by Adaptyv Bio (App. H.12). Among 28 expressed designs, Adaptyv Bio classified one as a binder based on a concentration-dependent SPR response with discernible association and dissociation phases in the sensorgram (App. Fig. 8).

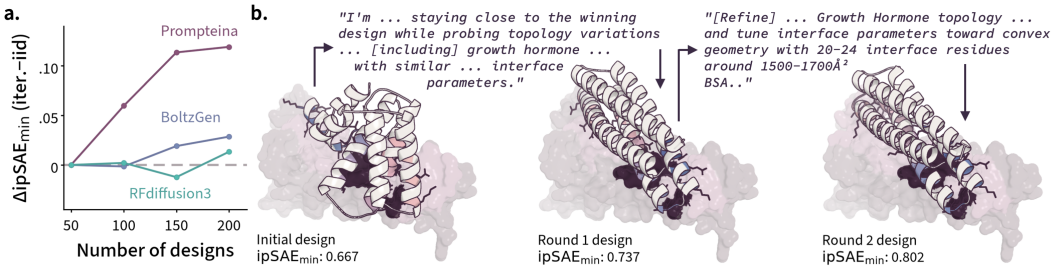


Figure 5: **Prompteina gains most from iterative optimization on Claudin-1.** **(a)** Difference in cumulative top-1 $\text{ipSAE}_{\text{min}}$ between iterative and matched-budget i.i.d. generation, using the same initial pool of 50 designs. **(b)** Planning excerpts and selected Claudin-1 designs from the first two feedback rounds. The agent explores alternative topologies in round 1 and enlarges the requested interface in round 2. For each complex, we report $\text{ipSAE}_{\text{min}}$ (↑) as a proxy for binding strength.

4 Related Work

Protein structure generation. Diffusion and flow-matching models progressed from backbone generation (Yim et al., 2023; Watson et al., 2023; Geffner et al., 2025) to all-atom methods that generate sequence and side-chain geometry (Geffner et al., 2026; Didi et al., 2026; Butcher et al., 2025; Rector-Brooks et al., 2026; Stark et al., 2026). RFdiffusion3 and BoltzGen combine many controls in one model, but each property has a distinct model input and its own training annotations (Butcher et al., 2025; Stark et al., 2026). ESM3 jointly generates sequence, structure, and function through dedicated token tracks (Hayes et al., 2025). Chroma provides the closest precedent for using natural language to control protein structure generation. It guides sampling through a separate structure-to-caption model rather than conditioning its denoiser on text (Ingraham et al., 2023). However, the authors report that natural-language conditioning remains sensitive to guidance strength.

Protein–text multimodal models. Protein–text models learn shared representations for retrieval and classification (Xu et al., 2023b; Su et al., 2025; Xu et al., 2023a), generate captions from sequence or structure (Liu et al., 2024; Fei et al., 2025; Abdine et al., 2024), and generate proteins from text. ProteinDT, Pinal, InstructPro, InstructBioMol, and BioMatrix generate amino-acid sequences, sometimes through a text-aligned latent or discrete structure representation (Liu et al., 2025; Dai et al., 2025; Song et al., 2026; Zhuang et al., 2025; Pei et al., 2026). HelixProtX emits backbone torsion angles from text, but its pretrained parameters are unavailable for evaluation (Chen et al., 2024). Prompteina pairs annotated text spans with residues and directly conditions a continuous all-atom structure generation model on global and token-level text representations (App. A).

5 Discussion

We introduce **Prompteina**, a text-conditioned all-atom protein structure generation model, and **Prompteina-Data**, a paired text–structure dataset of 1.8M structures and 52.4M captions. Prompteina uses a single natural-language interface to control global and residue-specific structural and functional properties. On properties that existing design models can express, Prompteina matches or exceeds dedicated conditioning approaches, while supporting new properties and a broader range than any single baseline. The same interface lets an LLM agent research a design objective and directly translate its conclusions into design prompts, rather than reduce them to the structured formats other models require. In protein binder design, Prompteina turns this reasoning into better designs than the same agent produces through existing design models. As LLM agents improve, the conditioning interfaces exposed by specialist generative models become increasingly important: agent reasoning can guide generation only when the model can accept the resulting design specifications. This same workflow also supports iterative design and optimization, where an agent can incorporate in silico or experimental feedback, update its design objectives, and refine subsequent designs. Together, these results show that natural language provides a general interface for connecting scientific reasoning to protein structure generation.

Ethics Statement

Prompteina is a general protein-generation system and therefore has potential dual-use implications. This work focuses on protein structure generation and binder design for research and therapeutic targets; it does not design or optimize pathogens, toxins, virulence, immune evasion, or host range. The GM2A experiment used an in vitro binding assay and did not involve infectious agents or live pathogens. We will release models and data with documentation of their intended use and limitations. The authors have read and adhere to the ICLR Code of Ethics.

Reproducibility Statement

To support reproducibility, we will release the data, code, and model weights used in this work. The paper and appendices document the data-processing, training, evaluation, and experimental procedures needed to reproduce the reported results.

AI Use Statement

In this work, we used generative AI tools to generate a synthetic caption dataset with Qwen3; conduct agentic binder-design experiments with Claude, including target research, epitope selection, design specification, and iterative optimization; assist with research execution, including writing code and running and monitoring experiments; and polish writing. We manually checked synthetic data for accurracy; self and peer-reviewed any AI-written code; and reviewed all experimental outputs. Generative AI was not used to directly draft sections of the paper or to formulate or prove mathematical claims. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims, code, data, and artifacts produced with the aid of generative AI.

References

Hadi Abdine, Michail Chatzianastasis, Costas Bouyioukos, and Michalis Vazirgiannis. Prot2Text: Multimodal Protein's Function Generation with GNNs and Transformers. *AAAI*, 38(10):10757–10765, March 2024. ISSN 2374-3468, 2159-5399. doi: 10.1609/aaai.v38i10.28948.

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J. Ballard, Joshua Bambrick, Sebastian W. Bodenstein, David A. Evans, Chia-Chun Hung, Michael O'Neill, David Reiman, Kathryn Tunyasuvunakool, Zachary Wu, Akvilė Žemgulytė, Eirini Arvaniti, Charles Beattie, Ottavia Bertolli, Alex Bridgland, Alexey Cherepanov, Miles Congreve, Alexander I. Cowen-Rivers, Andrew Cowie, Michael Figurnov, Fabian B. Fuchs, Hannah Gladman, Rishub Jain, Yousuf A. Khan, Caroline M. R. Low, Kuba Perlin, Anna Potapenko, Pascal Savy, Sukhdeep Singh, Adrian Stecula, Ashok Thillaisundaram, Catherine Tong, Sergei Yakneen, Ellen D. Zhong, Michal Zielinski, Augustin Žídek, Victor Bapst, Pushmeet Kohli, Max Jaderberg, Demis Hassabis, and John M. Jumper. Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature*, 630(8016):493–500, June 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07487-w.

Woody Ahern, Jason Yim, Doug Tischer, Saman Salike, Seth M. Woodbury, Donghyo Kim, Indrek Kalvet, Yakov Kipnis, Brian Coventry, Han Raut Altae-Tran, Magnus S. Bauer, Regina Barzilay, Tommi S. Jaakkola, Rohith Krishna, and David Baker. Atom-level enzyme active site scaffolding using RFdiffusion2. *Nat Methods*, 23(1):96–105, January 2026. ISSN 1548-7105. doi: 10.1038/s41592-025-02975-x.

Helen M. Berman, John Westbrook, Zukang Feng, Gary Gilliland, T. N. Bhat, Helge Weissig, Ilya N. Shindyalov, and Philip E. Bourne. The Protein Data Bank. *Nucleic Acids Res*, 28(1):235–242, January 2000. ISSN 0305-1048. doi: 10.1093/nar/28.1.235.

Jasper Butcher, Rohith Krishna, Raktim Mitra, Rafael I. Brent, Yanjing Li, Nathaniel Corley, Paul T. Kim, Jonathan Funk, Simon Mathis, Saman Salike, Aiko Muraishi, Helen Eisenach, Tuscan Rock Thompson, Jie Chen, Yuliya Politanska, Enisha Sehgal, Brian Coventry, Odin Zhang, Bo Qiang, Kieran Didi, Max Kazman, Frank DiMaio, and David Baker. De novo Design of All-atom Biomolecular Interactions with RFdiffusion3, November 2025. ISSN 2692-8205.

Longxing Cao, Brian Coventry, Inna Goreshnik, Buwei Huang, William Sheffler, Joon Sung Park, Kevin M. Jude, Iva Marković, Rameshwar U. Kadam, Koen H. G. Verschueren, Kenneth Verstraete, Scott Thomas Russell Walsh, Nathaniel Bennett, Ashish Phal, Aerin Yang, Lisa Kozodoy, Michelle DeWitt, Lora Picton, Lauren Miller, Eva-Maria Strauch, Nicholas D. DeBouver, Allison Pires, Asim K. Bera, Samer Halabiya, Bradley Hammerson, Wei Yang, Steffen Bernard, Lance Stewart, Ian A. Wilson, Hannele Ruohola-Baker, Joseph Schlessinger, Sangwon Lee, Savvas N. Savvides, K. Christopher Garcia, and David Baker. Design of protein-binding proteins from the target structure alone. *Nature*, 605(7910):551–560, May 2022. ISSN 1476-4687. doi: 10.1038/s41586-022-04654-9.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. PixArt-α: Fast Training of Diffusion Transformer for Photorealistic Text-to-Image Synthesis, December 2023.

Zhiyuan Chen, Tianhao Chen, Chenggang Xie, Yang Xue, Xiaonan Zhang, Jingbo Zhou, and Xiaomin Fang. Unifying Sequences, Structures, and Descriptions for Any-to-Any Protein Generation with the Large Multimodal Model HelixProtX, July 2024.

Hua Cheng, R. Dustin Schaeffer, Yuxing Liao, Lisa N. Kinch, Jimin Pei, Shuoyong Shi, Bong-Hyun Kim, and Nick V. Grishin. ECOD: An Evolutionary Classification of Protein Domains. *PLOS Computational Biology*, 10(12):e1003926, December 2014. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1003926.

Pawel Dabrowski-Tumanski, Pawel Rubach, Dimos Goundaroulis, Julien Dorier, Piotr Sułkowski, Kenneth C Millett, Eric J Rawdon, Andrzej Stasiak, and Joanna I Sulkowska. KnotProt 2.0: A database of proteins with knots and other entangled structures. *Nucleic Acids Res*, 47(D1): D367–D375, January 2019. ISSN 0305-1048. doi: 10.1093/nar/gky1140.

Fengyuan Dai, Shiyang You, Yudian Zhu, Yuan Gao, Lihao Fu, Xibin Zhou, Jin Su, Chentong Wang, Yuliang Fan, Xiaoxiao Ma, Xianjun Deng, Letong Yu, Hui Qian, Yan He, Yitao Ke, Chenchen Han, Xing Chang, Liangzhen Zheng, Sheng Wang, Yajie Wang, Anping Zeng, Shunzhi Wang, Tong Si, Jianming Liu, Hongyuan Lu, and Fajie Yuan. Toward De Novo Protein Design from Natural Language, September 2025. ISSN 2692-8205.

Jose M Dana, Aleksandras Gutmanas, Nidhi Tyagi, Guoying Qi, Claire O'Donovan, Maria Martin, and Sameer Velankar. SIFTS: Updated Structure Integration with Function, Taxonomy and Sequences resource allows 40-fold increase in coverage of structure-based annotations for proteins. *Nucleic Acids Res*, 47(D1):D482–D489, January 2019. ISSN 0305-1048. doi: 10.1093/nar/gky1114.

J. Dauparas, I. Anishchenko, N. Bennett, H. Bai, R. J. Ragotte, L. F. Milles, B. I. M. Wicky, A. Courbet, R. J. de Haas, N. Bethel, P. J. Y. Leung, T. F. Huddy, S. Pellock, D. Tischer, F. Chan, B. Koepnick, H. Nguyen, A. Kang, B. Sankaran, A. K. Bera, N. P. King, and D. Baker. Robust deep learning–based protein sequence design using ProteinMPNN. *Science*, 378(6615):49–56, October 2022. doi: 10.1126/science.add2187.

Natalie M. Deschenes, Camilyn Cheng, Alex E. Ryckman, Brianna M. Quinville, Prem Khanal, Melissa Mitchell, Zhilin Chen, Waheed Sangrar, Steven J. Gray, and Jagdeep S. Walia. Biochemical Correction of GM2 Ganglioside Accumulation in AB-Variant GM2 Gangliosidosis. *International Journal of Molecular Sciences*, 24(11):9217, January 2023. ISSN 1422-0067. doi: 10.3390/ijms24119217.

Kieran Didi, Zuobai Zhang, Guoqing Zhou, Danny Reidenbach, Zhonglin Cao, Sooyoung Cha, Tomas Geffner, Christian Dallago, Jian Tang, Michael M. Bronstein, Martin Steinegger, Emine Kucukbenli, Arash Vahdat, and Karsten Kreis. Scaling Atomistic Protein Binder Design with Generative Pretraining and Test-Time Compute, March 2026.

James Dunbar, Konrad Krawczyk, Jinwoo Leem, Terry Baker, Angelika Fuchs, Guy Georges, Jiye Shi, and Charlotte M. Deane. SAbDab: The structural antibody database. *Nucleic Acids Res*, 42 (D1):D1140–D1146, January 2014. ISSN 0305-1048. doi: 10.1093/nar/gkt1043.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling Rectified Flow Transformers for High-Resolution Image Synthesis, March 2024.

Xiao Fei, Michail Chatzianastasis, Sarah Almeida Carneiro, Hadi Abdine, Lawrence P. Petalidis, and Michalis Vazirgiannis. Prot2Text-V2: Protein Function Prediction with Multimodal Contrastive Alignment, October 2025.

Tomas Geffner, Kieran Didi, Zuobai Zhang, Danny Reidenbach, Zhonglin Cao, Jason Yim, Mario Geiger, Christian Dallago, Emine Kucukbenli, Arash Vahdat, and Karsten Kreis. Proteina: Scaling Flow-based Protein Structure Generative Models, March 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, May 2026.

Alireza Ghafarollahi and Markus J. Buehler. ProtAgents: Protein discovery via large language model multi-agent collaborations combining physics and machine learning. *Digital Discovery*, 3 (7):1389–1409, July 2024. ISSN 2635-098X. doi: 10.1039/d4dd00013g.

Thomas Hayes, Roshan Rao, Halil Akin, Nicholas J. Sofroniew, Deniz Oktay, Zeming Lin, Robert Verkuil, Vincent Q. Tran, Jonathan Deaton, Marius Wiggert, Rohil Badkundri, Irhum Shafkat, Jun Gong, Alexander Derry, Raul S. Molina, Neil Thomas, Yousuf A. Khan, Chetan Mishra, Carolyn Kim, Liam J. Bartie, Matthew Nemeth, Patrick D. Hsu, Tom Sercu, Salvatore Candido, and Alexander Rives. Simulating 500 million years of evolution with a language model. *Science*, 387(6736):850–858, February 2025. doi: 10.1126/science.ads0018.

Maarten L. Hekkelman, Daniel Álvarez Salmoral, Anastassis Perrakis, and Robbie P. Joosten. DSSP 4: FAIR annotation of protein secondary structure. *Protein Sci*, 34(8):e70208, July 2025. ISSN 0961-8368. doi: 10.1002/pro.70208.

Jonathan Ho and Tim Salimans. Classifier-Free Diffusion Guidance, July 2022.

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, Shan Tie, Vincent Xue, Sarah C. Cowles, Alan Leung, João V. Rodrigues, Claudio L. Morales-Perez, Alex M. Ayoub, Robin Green, Katherine Puentes, Frank Oplinger, Nishant V. Panwar, Fritz Obermeyer, Adam R. Root, Andrew L. Beam, Frank J. Poelwijk, and Gevorg Grigoryan. Illuminating protein space with a programmable generative model. *Nature*, 623(7989):1070–1078, November 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-06728-8.

S Jones and J M Thornton. Principles of protein-protein interactions. *Proceedings of the National Academy of Sciences*, 93(1):13–20, January 1996. doi: 10.1073/pnas.93.1.13.

W. Kabsch and C. Sander. Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22(12):2577–2637, December 1983. ISSN 0006-3525. doi: 10.1002/bip.360221211.

Dániel Kozma, István Simon, and Gábor E. Tusnády. PDBTM: Protein Data Bank of transmembrane proteins after 8 years. *Nucleic Acids Res*, 41(Database issue):D524–D529, January 2013. ISSN 0305-1048. doi: 10.1093/nar/gks1169.

Yeqing Lin and Mohammed AlQuraishi. Generating Novel, Designable, and Diverse Protein Structures by Equivariantly Diffusing Oriented Residue Clouds, June 2023.

Yeqing Lin, Minji Lee, Zhao Zhang, and Mohammed AlQuraishi. Out of Many, One: Designing and Scaffolding Proteins at the Scale of the Structural Universe with Genie 2, May 2024.

Yeqing Lin, Minji Lee, Aakarsh Vermani, Ellena Jiang, Sebastiaan De Cooman, Matej Špeťko, and Mohammed AlQuraishi. Fast and Ultra-Capable Protein Design: Advancing the Frontier Through Atomistic SE(3)-Equivariance with Genie 3, May 2026. ISSN 2692-8205.

Shengchao Liu, Yanjing Li, Zhuoxinran Li, Anthony Gitter, Yutao Zhu, Jiarui Lu, Zhao Xu, Weili Nie, Arvind Ramanathan, Chaowei Xiao, Jian Tang, Hongyu Guo, and Anima Anandkumar. A text-guided protein design framework. *Nat Mach Intell*, 7(4):580–591, April 2025. ISSN 2522-5839. doi: 10.1038/s42256-025-01011-z.

Zhiyuan Liu, An Zhang, Hao Fei, Enzhi Zhang, Xiang Wang, Kenji Kawaguchi, and Tat-Seng Chua. ProtT3: Protein-to-Text Generation for Text-based Protein Understanding, May 2024.

Douglas Myers-Turnbull, Spencer E. Bliven, Peter W. Rose, Zaid K. Aziz, Philippe Youkharibache, Philip E. Bourne, and Andreas Prlić. Systematic detection of internal symmetry in proteins using CE-Symm. *J Mol Biol*, 426(11):2255–2268, May 2014. ISSN 1089-8638. doi: 10.1016/j.jmb.2014.03.010.

Max D. Overath, Andreas S. H. Rygaard, Christian P. Jacobsen, Valentas Brasas, Oliver Morell, Pietro Sormanni, and Timothy P. Jenkins. Predicting Experimental Success in De Novo Binder Design: A Meta-Analysis of 3,766 Experimentally Characterised Binders, September 2025. ISSN 2692-8205.

Martin Pacesa, Lennart Nickel, Christian Schellhaas, Joseph Schmidt, Ekaterina Pyatova, Lucas Kissling, Patrick Barendse, Jagrity Choudhury, Srajan Kapoor, Ana Alcaraz-Serna, Yehlin Cho, Kourosh H. Ghamary, Laura Vinué, Brahm J. Yachnin, Andrew M. Wollacott, Stephen Buckley, Adrie H. Westphal, Simon Lindhoud, Sandrine Georgeon, Casper A. Goverde, Georgios N. Hatzopoulos, Pierre Gönczy, Yannick D. Muller, Gerald Schwank, Daan C. Swarts, Alex J. Vecchio, Bernard L. Schneider, Sergey Ovchinnikov, and Bruno E. Correia. One-shot design of functional protein binders with BindCraft. *Nature*, 646(8084):483–492, October 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09429-6.

Lisanna Paladin, Martina Bevilacqua, Sara Errigo, Damiano Piovesan, Ivan Mičetić, Marco Necci, Alexander Miguel Monzon, Maria Laura Fabre, Jose Luis Lopez, Juliet F Nilsson, Javier Rios, Pablo Lorenzano Menna, Maia Cabrera, Martin Gonzalez Buitron, Mariane Gonçalves Kulik, Sebastian Fernandez-Alberti, Maria Silvina Fornasari, Gustavo Parisi, Antonio Lagares, Layla Hirsh, Miguel A Andrade-Navarro, Andrey V Kajava, and Silvio C E Tosatto. RepeatsDB in 2021: Improved data and extended classification for protein tandem repeat structures. *Nucleic Acids Res*, 49(D1):D452–D457, January 2021. ISSN 0305-1048. doi: 10.1093/nar/gkaa1097.

William Peebles and Saining Xie. Scalable Diffusion Models with Transformers, March 2023.

Qizhi Pei, Zhimeng Zhou, Yi Duan, Yiyang Zhao, Wei Li, Han Guo, Liang He, Chengping Li, Chang-Yu Hsieh, Conghui He, Rui Yan, and Lijun Wu. BioMatrix: Towards a Comprehensive Biological Foundation Model Spanning the Modality Matrix of Sequences, Structures, and Language, June 2026.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis, July 2023.

Ranjit Prasad Bahadur, Pinak Chakrabarti, Francis Rodier, and Joël Janin. A Dissection of Specific and Non-specific Protein–Protein Interfaces. *Journal of Molecular Biology*, 336(4):943–955, February 2004. ISSN 0022-2836. doi: 10.1016/j.jmb.2003.12.073.

Lukáš Pravda, David Sehnal, Radka Svobodová Vařeková, Veronika Navrátilová, Dominik Toušek, Karel Berka, Michal Otyepka, and Jaroslav Koča. ChannelsDB: Database of biomacromolecular tunnels and pores. *Nucleic Acids Res*, 46(D1):D399–D405, January 2018. ISSN 0305-1048. doi: 10.1093/nar/gkx868.

Yanru Qu, Cheng-Yen Hsieh, Zaixiang Zheng, Ge Liu, and Quanquan Gu. Protein Autoregressive Modeling via Multiscale Structure Generation, May 2026.

Jarrid Rector-Brooks, Théophile Lambert, Marta Skreta, Daniel Roth, Yueming Long, Zi-Qi Li, Xi Zhang, Miruna Cretu, Francesca-Zhoufan Li, Tanvi Ganapathy, Emily Jin, Avishek Joey Bose, Jason Yang, Kirill Neklyudov, Yoshua Bengio, Alexander Tong, Frances H. Arnold, and Cheng-Hao Liu. General Multimodal Protein Design Enables DNA-Encoding of Chemistry, April 2026.

Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative Adversarial Text to Image Synthesis, June 2016.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis with Latent Diffusion Models, April 2022.

Burkhard Rost. Twilight zone of protein sequence alignments. *Protein Engineering*, 12(2):85–94, February 1999. ISSN 0269-2139. doi: 10.1093/protein/12.2.85.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding, May 2022.

Amir Shanehsazzadeh. Autonomous De Novo Protein Binder Design with Claude, 2026.

Ian Sillitoe, Tony E. Lewis, Alison Cuff, Sayoni Das, Paul Ashford, Natalie L. Dawson, Nicholas Furnham, Roman A. Laskowski, David Lee, Jonathan G. Lees, Sonja Lehtinen, Romain A. Studer, Janet Thornton, and Christine A. Orengo. CATH: Comprehensive structural and functional annotations for genome sequences. *Nucleic Acids Res*, 43(D1):D376–D381, January 2015. ISSN 0305-1048. doi: 10.1093/nar/gku947.

Zhenqiao Song, Ramith Hettiarachchi, Chuan Li, Jianwen Xie, and Lei Li. InstructPro: Natural Language Guided Ligand-Binding Protein Design, March 2026.

Hannes Stark, Felix Faltings, MinGyu Choi, Yuxin Xie, Eunsu Hur, Timothy O'Donnell, Anton Bushuiev, Talip Uçar, Saro Passaro, Weian Mao, Mateo Reveiz, Roman Bushuiev, Tally Portnoi, 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, Samuel J. Snyder, A. Katherine Hatstat, Angelika Arada, Nam Hyeong Kim, Haoyu Fan, 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, June 2026. ISSN 2692-8205.

Jin Su, Yan He, Shiyang You, Shiyu Jiang, Xibin Zhou, Xuting Zhang, Yuxuan Wang, Xining Su, Igor Tolstoy, Xing Chang, Hongyuan Lu, and Fajie Yuan. A trimodal protein language model enables advanced protein searches. *Nat Biotechnol*, pp. 1–7, October 2025. ISSN 1546-1696. doi: 10.1038/s41587-025-02836-0.

The Gene Ontology Consortium. The Gene Ontology resource: Enriching a GOld mine. *Nucleic Acids Res*, 49(D1):D325–D334, January 2021. ISSN 0305-1048. doi: 10.1093/nar/gkaa1113.

The UniProt Consortium. UniProt: The Universal Protein Knowledgebase in 2025. *Nucleic Acids Res*, 53(D1):D609–D617, January 2025. ISSN 1362-4962. doi: 10.1093/nar/gkae1010.

Michel van Kempen, Stephanie S. Kim, Charlotte Tumescheit, Milot Mirdita, Jeongjae Lee, Cameron L. M. Gilchrist, Johannes Söding, and Martin Steinegger. Fast and accurate protein structure search with Foldseek. *Nat Biotechnol*, 42(2):243–246, February 2024. ISSN 1546-1696. doi: 10.1038/s41587-023-01773-0.

Mihaly Varadi, Stephen Anyango, Mandar Deshpande, Sreenath Nair, Cindy Natassia, Galabina Yordanova, David Yuan, Oana Stroe, Gemma Wood, Agata Laydon, Augustin Žídek, Tim Green, Kathryn Tunyasuvunakool, Stig Petersen, John Jumper, Ellen Clancy, Richard Green, Ankur Vora, Mira Lutfi, Michael Figurnov, Andrew Cowie, Nicole Hobbs, Pushmeet Kohli, Gerard Kleywegt, Ewan Birney, Demis Hassabis, and Sameer Velankar. AlphaFold Protein Structure Database: Massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic Acids Res*, 50(D1):D439–D444, January 2022. ISSN 0305-1048. doi: 10.1093/nar/gkab1061.

Zirui Wang, Zhizhou Sha, Zheng Ding, Yilin Wang, and Zhuowen Tu. TokenCompose: Text-to-Image Diffusion with Token-level Supervision, June 2024.

Joseph L. Watson, David Juergens, Nathaniel R. Bennett, Brian L. Trippe, Jason Yim, Helen E. Eisenach, Woody Ahern, Andrew J. Borst, Robert J. Ragotte, Lukas F. Milles, Basile I. M. Wicky, Nikita Hanikel, Samuel J. Pellock, Alexis Courbet, William Sheffler, Jue Wang, Preetham Venkatesh, Isaac Sappington, Susana Vázquez Torres, Anna Lauko, Valentin De Bortoli, Emile Mathieu, Sergey Ovchinnikov, Regina Barzilay, Tommi S. Jaakkola, Frank DiMaio, Minkyung Baek, and David Baker. De novo design of protein structure and function with RFdiffusion. *Nature*, 620(7976):1089–1100, August 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-06415-8.

Jude Wells, Alex Hawkins-Hooker, Nicola Bordin, Ian Sillitoe, Brooks Paige, and Christine Orengo. Chainsaw: Protein domain segmentation with fully convolutional neural networks. *Bioinformatics*, 40(5):btae296, May 2024. ISSN 1367-4811. doi: 10.1093/bioinformatics/btae296.

Henry R. Wilman, Jiye Shi, and Charlotte M. Deane. Helix kinks are equally prevalent in soluble and membrane proteins. *Proteins: Structure, Function, and Bioinformatics*, 82(9):1960–1970, 2014. ISSN 1097-0134. doi: 10.1002/prot.24550.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng, Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. Qwen-Image Technical Report, August 2025.

Hanwen Xu, Addie Woicik, Hoifung Poon, Russ B. Altman, and Sheng Wang. Multilingual translation for zero-shot biomedical classification using BioTranslator. *Nat Commun*, 14(1):738, February 2023a. ISSN 2041-1723. doi: 10.1038/s41467-023-36476-2.

Minghao Xu, Xinyu Yuan, Santiago Miret, and Jian Tang. ProtST: Multi-Modality Learning of Protein Sequences and Biomedical Texts, July 2023b.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 Technical Report, May 2025.

Jason Yim, Brian L. Trippe, Valentin De Bortoli, Emile Mathieu, Arnaud Doucet, Regina Barzilay, and Tommi Jaakkola. SE(3) diffusion model with application to protein backbone generation, May 2023.

Vinicius Zambaldi, David La, Alexander E. Chu, Harshnira Patani, Amy E. Danson, Tristan O. C. Kwan, Thomas Frerix, Rosalia G. Schneider, David Saxton, Ashok Thillaisundaram, Zachary Wu, Isabel Moraes, Oskar Lange, Eliseo Papa, Gabriella Stanton, Victor Martin, Sukhdeep Singh, Lai H. Wong, Russ Bates, Simon A. Kohl, Josh Abramson, Andrew W. Senior, Yilmaz Alguel, Mary Y. Wu, Irene M. Aspalter, Katie Bentley, David L. V. Bauer, Peter Cherepanov, Demis Hassabis, Pushmeet Kohli, Rob Fergus, and Jue Wang. De novo design of high-affinity protein binders with AlphaProteo. https://arxiv.org/abs/2409.08022v1, September 2024.

Yang Zhang and Jeffrey Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins*, 57(4):702–710, December 2004. ISSN 1097-0134. doi: 10.1002/prot.20264.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models, June 2025.

Zuobai Zhang, Minghao Xu, Arian Jamasb, Vijil Chenthamarakshan, Aurelie Lozano, Payel Das, and Jian Tang. Protein Representation Learning by Geometric Structure Pretraining, January 2023.

Xiang Zhuang, Keyan Ding, Tianwen Lyu, Yinuo Jiang, Xiaotong Li, Zhuoyi Xiang, Zeyuan Wang, Ming Qin, Kehua Feng, Jike Wang, Qiang Zhang, and Huajun Chen. Advancing biomolecular understanding and design following human instructions. *Nat Mach Intell*, 7(7):1154–1167, July 2025. ISSN 2522-5839. doi: 10.1038/s42256-025-01064-0.

APPENDIX CONTENTS

A Extended related work **19**

B Prompteina-Data construction **22**

B.1 Data sources and scale 22

B.2 Annotation extraction and categories 22

B.3 Residue-specific captions 23

B.4 Caption sampling 23

B.5 LLM caption generation 24

B.6 Text–residue alignment annotations 25

C Model and training details **26**

C.1 Inherited Complexa model 26

C.2 Text encoders 26

C.3 Token cross-attention 26

C.4 Text–residue alignment head 27

C.5 Initialization and gating 27

C.6 Training 27

C.7 Sampling 29

D Ablations **30**

E Monomer generation evaluation **32**

E.1 Prompt paraphrase robustness 34

E.2 Designability 34

E.3 Structural novelty 36

E.4 Full prompts for Figure 1a 37

F CATH verifier calibration **38**

G Functional specification evaluation **39**

H Binder design evaluation **41**

H.1 Interface-residue steering 41

H.2 Agentic binder design protocol 41

H.3 Baseline agent specifications 46

H.4 Binder score calibration 46

H.5 Prediction-model replication 48

H.6 Predicted binding across design budgets 48

H.7 Generated binder–target interface properties 48

H.8 Binder specification ablation 50

H.9 Structural diversity 50

H.10 Iterative prompt optimization 52

H.11 Full-panel iterative prompt optimization 53
H.12 Prospective GM2A binding assay 55

A EXTENDED RELATED WORK

Recent protein structure generation models control different properties through distinct architectural components, training procedures, and inference-time operations. We survey these mechanisms before describing the text-conditioning techniques that Prompteina adopts from image generation. The comparison focuses on the dedicated input formats and procedures that existing models require for each controlled property.

RFdiffusion (Watson et al., 2023). RFdiffusion fine-tunes a pretrained RoseTTAFold structure prediction network as a denoiser for the reverse diffusion process. Five separately fine-tuned variants handle unconditional generation, fold-conditioned generation, binder design, fold-conditioned binder design, and enzyme active-site scaffolding. Fold control takes per-residue secondary-structure labels and a residue-by-residue block adjacency matrix that specifies long-range contacts. Binder design takes a user-specified subset of target-chain residues as interface hotspots and can combine them with fold properties to control binder scaffold topology. Motif scaffolding fixes the sequence identity and three-dimensional coordinates of motif backbone and side-chain atoms during denoising. Symmetric oligomer design takes a cyclic, dihedral, tetrahedral, octahedral, or icosahedral point group and initializes noise with the corresponding symmetry. Each task therefore uses a distinct input format and separately trained model.

RFdiffusion2 (Ahern et al., 2026). RFdiffusion2 uses the RoseTTAFold All-Atom architecture and combines Euclidean flow matching for atom coordinates with Riemannian flow matching on $\mathrm{SE}(3)$ for backbone frames. The model is trained from random initialization and focuses on enzyme active-site scaffolding. Atomic motif conditioning supplies individual heavy atoms from catalytic side chains. During training, a random residue subset forms the motif, and each selected residue is represented either as a backbone frame or as an atomized subgraph of its heavy atoms. Unindexed motif conditioning removes sequence indices from these residues, requiring the model to infer their positions in the scaffold. Relative accessible surface area (RASA) labels mark each ligand atom as buried or exposed during 50% of training. An origin pseudo-atom gives an approximate center-of-mass target for the scaffold relative to the ligand. Stochastic centering prevents the model from recovering the motif–scaffold offset directly from partially noised inputs by adding a small Gaussian translation to the training structure. A single model handles these tasks, but each property has a dedicated input format and training procedure.

RFdiffusion3 (Butcher et al., 2025). RFdiffusion3 uses a transformer-based U-Net for all-atom protein generation. A single model handles unconditional generation, protein, nucleic-acid, and small-molecule binder design, enzyme active-site scaffolding, and symmetric oligomer design through several conditioning mechanisms. Atom-level features label target or ligand atoms with property-specific annotations at varying rates during training. Donor and acceptor labels direct complementary hydrogen bonds, RASA labels divide ligand-atom solvent accessibility into four burial bins, and hotspots mark target atoms near the requested binder–target interface. Appended tokens represent catalytic residues for unindexed motif scaffolding, leaving the model to infer each catalytic residue's sequence position. Inference-time operations symmetrize each diffusion step for oligomer design and position the generated protein relative to the target center of mass. For nucleic-acid and small-molecule targets, RFdiffusion3 can jointly diffuse the target conformation with the protein. The shared model still requires a distinct input format and annotation pipeline for each property.

Proteina series (Geffner et al., 2025; 2026; Didi et al., 2026). Proteina (Geffner et al., 2025) is a non-equivariant flow-matching model for protein backbone generation. CATH fold conditioning uses learnable embeddings for a hierarchy of 5 classes, 43 architectures, and 1,336 topologies, injected through adaptive layer normalization (AdaLN). Hierarchical dropout trains unconditional generation and each level of the hierarchy. The model drops all labels with probability 0.5, retains only class with probability 0.1, retains class and architecture with probability 0.15, and retains all three levels with probability 0.25. La-Proteina (Geffner et al., 2026) extends Proteina to all-atom generation through a partially latent representation and adds indexed and unindexed motif scaffolding, but omits fold conditioning. Complexa (Didi et al., 2026) extends La-Proteina to binder design with separate protein–protein, CATH-conditioned protein–protein, and ligand–protein checkpoints. Protein and small-molecule targets use different representation pipelines. Binder design takes target

interface hotspots, and the CATH-conditioned checkpoint uses Proteina's hierarchical embeddings and AdaLN but applies only the class level at inference.

Chroma (Ingraham et al., 2023). Chroma composes a library of separate conditioners at sampling time. Each independently trained or hand-designed conditioner steers generation through its gradient with respect to the noised structure. Symmetry conditioning tessellates an asymmetric subunit into C_n, D_n, T, O, or I point groups, and shape conditioning minimizes an optimal-transport distance to a user-provided point cloud. Fold-class conditioning inverts a separately trained CATH classifier through its gradients. Substructure conditioning fixes motif coordinates during registered or unregistered scaffolding. Text conditioning trains a structure-to-caption predictor and uses the gradient of caption log-likelihood to steer generation toward the prompt. Guidance strength must be balanced against the prior: weak gradients have little effect, whereas strong gradients overwhelm the prior and degrade structural coherence. The authors report that guidance effects vary widely and that robust natural-language conditioning remains challenging. The ProCap captioner pairs a frozen structure encoder with GPT-Neo 125M and is trained on approximately 45,000 captions from PDB descriptions and UniProt functional comments. The authors report that this dataset size constrained their architecture choices. Any subset of conditioners can be composed at inference, but each requires a trained classifier, analytical energy function, or separate gradient computation.

HelixProtX (Chen et al., 2024). HelixProtX is a multimodal language model that maps among protein sequence, structure, and text. Generating structure from text emits backbone torsion angles, and control is reported at the level of the whole structure rather than individual residues or motifs. Its released repository does not include the pretrained HelixProtX parameters, which prevents evaluation on our benchmark.[1]

DISCO (Rector-Brooks et al., 2026). DISCO jointly generates protein sequences and structures with Euclidean coordinate diffusion and absorbing-state masked sequence diffusion. Biomolecular context enters as discrete tokens with three-dimensional coordinates for small molecules, metallocofactors, reactive intermediates, or nucleic acids, which are denoised with the protein. Property-specific control uses Feynman–Kac reward tilting, with a custom reward function, distance thresholds, and hyperparameters for properties such as disulfides or cation–π interactions. Target-specific binding guidance evaluates two aligned denoiser calls per timestep and requires a hyperparameter sweep for each target pair.

PAR (Qu et al., 2026). PAR uses an autoregressive model and flow-matching decoder to generate protein backbones by progressively upsampling a coarse representation. Its two conditioning mechanisms operate on this hierarchy without fine-tuning. Structural prompting replaces the coarsest generation step with user-provided three-dimensional points that define an approximate layout, which PAR then upsamples into a full backbone. Motif scaffolding restores the ground-truth coordinates of motif residues after every prediction scale, keeping the motif fixed as the surrounding scaffold is generated.

BoltzGen (Stark et al., 2026). BoltzGen extends Boltz-2 into a single model for folding, unconditional design, binder design, motif scaffolding, and non-protein interface design through seven training tasks. Four conditioning mechanisms use separate representations and training distributions. Binding-site conditioning assigns binding, non-binding, or unspecified labels to target residues. Structure groups use a distance matrix to preserve pairwise distances among residues in the same group. Secondary-structure conditioning labels each designed residue as helix, strand, coil, or unspecified. Covalent-bond conditioning supplies an atom-pair bond matrix for cyclic proteins, disulfide staples, and helicons. Each property has its own representation, dropout scheme, and training-time sampling distribution.

ESM3 (Hayes et al., 2025). ESM3 is a multimodal masked language model that generates protein sequence, structure, and function, with each modality or property represented by a dedicated token track. For conditioning structure generation, partial information can be provided on any combination of property tracks: per-residue secondary structure labels, solvent-accessible surface area bins,

[1] HelixProtX repository.

function keywords derived from InterPro and Gene Ontology terms, residue-level functional annotations, partial structure (as discrete tokens or atomic coordinates), and partial sequence. Each track requires its own tokenization: structure uses a VQ-VAE with a 4096-entry codebook, function keywords require parsing free text into a 68,103-entry vocabulary, quantizing via 8 locality-sensitive hash functions, and a dedicated 3-layer transformer to invert the hashing. Each track also has its own noise schedule and dropout rate. Adding a new conditioning property requires designing a tokenization scheme, selecting a noise schedule, and training any auxiliary encoders or decoders the representation needs. Chroma and RFdiffusion already represent ESM3's per-residue secondary-structure input in the property-specific comparison, so we omit a redundant ESM3 run.

Genie series (Lin & AlQuraishi, 2023; Lin et al., 2024; 2026). Genie (Lin & AlQuraishi, 2023) generates protein backbones by diffusing C_α coordinates and conditions only on protein length. Genie 2 (Lin et al., 2024) introduces motif scaffolding by training entirely on scaffolding tasks. It represents motif sequence as one-hot vectors in the single-residue features and motif structure as $\mathrm{SE}(3)$-invariant pairwise C_α distances. This representation leaves Genie 2 to infer the relative configuration of multiple motifs. Genie 3 (Lin et al., 2026) uses a branched-polymer representation for all-atom generation and adds binder design. Its training distribution allocates 90% of examples to motif scaffolding and 10% to binder design, with subtasks that vary the motif and binder information supplied. Binder design takes target interface hotspots and can expand them to nearby surface residues or iteratively add residues from successful prior designs.

Text conditioning for diffusion models. Protein structure generation models increasingly use transformer backbones also found in text-conditioned image models (Geffner et al., 2025; Didi et al., 2026; Butcher et al., 2025; Rector-Brooks et al., 2026; Stark et al., 2026). That literature supplies the components Prompteina uses for natural-language conditioning. Frozen pretrained language models encode the text (Saharia et al., 2022), and later systems combine encoders to capture complementary information (Podell et al., 2023; Esser et al., 2024). Cross-attention carries token representations into the denoiser (Rombach et al., 2022), and adaptive layer normalization supplies a pooled representation (Peebles & Xie, 2023). PixArt-α combines these pathways and first trains the text projection while freezing the pretrained generation model (Chen et al., 2023). Classifier-free guidance trains conditional and unconditional predictions by dropping the conditioning on a fixed fraction of examples, then scales their difference at inference (Ho & Salimans, 2022). Prompteina adapts these components to protein structure generation.

B PROMPTEINA-DATA CONSTRUCTION

B.1 DATA SOURCES AND SCALE

We construct the paired text–structure dataset from three sources. Monomer training uses experimental protein chains from the Protein Data Bank (PDB) and predicted monomers from the AlphaFold Database (AFDB). Binder training uses Teddymer dimers assembled from TED domain pairs (Didi et al., 2026). The three corpora provide 1.8M entries and 52.4M captions (Table 4).

Table 4: **Training data composition.** Entries count independently captioned protein chains.

Corpus	Entries	Captions	Properties
PDB chains	611K	19.5M	59
AFDB monomers	288.7K	7.5M	26
Teddymer chains	903K	25.4M	32
Total	1.8M	52.4M	–

PDB chains include all protein chains (monomer and multimer) with no length or resolution filter at the captioning stage. At training time, chains are filtered to remove nucleic acid leaks, C_α-only entries, entries with fewer than 33 resolved residues, and chains longer than 512 residues. AFDB monomers are filtered by pLDDT confidence ($>$70) and maximum length ($<$512 residues), matching the base model training data. Teddymer dimers are synthetic complexes constructed by pairing TED-domain structures (Didi et al., 2026). We caption each chain independently, producing two caption sets per dimer. During binder training, each binder crop is conditioned on the caption for the chain selected as the binder.

B.2 ANNOTATION EXTRACTION AND CATEGORIES

Before captioning, we extract chain- or complex-level annotations from source databases and compute additional properties from atomic coordinates. The extraction produces 59 property types for PDB, 26 for AFDB, and 32 for Teddymer, organized into ten groups. Unless noted, a property appears in all three corpora. For AFDB, we use TED-derived CATH fold labels in place of experimental CATH assignments and exclude properties requiring experimental ligands, biological assemblies, or ECOD domain annotations. Teddymer is captioned per chain and excludes ligand, sheet topology, and PDB-specific properties.

- Fold and topology (11): CATH (Sillitoe et al., 2015) class, architecture, and topology; ECOD (Cheng et al., 2014) architecture, X-group, H-group, and T-group (PDB); contact order; internal symmetry detected via CE-Symm (Myers-Turnbull et al., 2014) (PDB); sheet topology including β-sandwich detection via SVD sheet-normal comparison (PDB, AFDB); knot topology from KnotProt (Dabrowski-Tumanski et al., 2019) (PDB).
- Secondary structure (3): helix count with an α, 3_{10}, and π breakdown, strand count, and sheet count (PDB, AFDB). We assign secondary structure with mkdssp (Hekkelman et al., 2025), merging helix gaps with `max_gap=` 2 to tolerate kinks (Wilman et al., 2014).
- Tertiary structural motifs (7): disulfide bonds with sequential separation class (local: $<$30 residues, medium-range: 30–80, long-range: $>$80) and disulfide topology (e.g., cystine knot, ladder, nested); metal coordination with geometry and ligand composition (e.g., tetrahedral zinc with 3 His, 1 Cys) (PDB); intra-chain salt bridges classified as buried or surface by relative solvent accessibility (PDB, AFDB); hydrogen-bond networks (PDB, AFDB); cis-peptides; β-turns.
- Global shape and surface (9): compactness with shape classification from SVD on C_α coordinates (spherical: elongation $<$1.5, elongated: 1.5–3.0, rod-like: $>$3.0, where elongation is the ratio of the two largest singular values); surface planarity; spatial aggregation propensity; buried unsatisfied hydrogen bonds; flexibility; hinge residues; repeat detection with secondary-structure composition; mechanical clamp type; Ramachandran profile.
- Quaternary structure (3): biological assembly symmetry (PDB), subunit structure (PDB), oligomeric context (PDB, Teddymer).

- Protein–protein interfaces (9; PDB, Teddymer): interface residues, buried surface area, hydrogen bonds (backbone/sidechain split), salt bridges, hydrophobic contacts, interface geometry, shape complementarity, secondary-structure character, calcium-dependent interfaces.
- Ligand binding (3): ligand identity with molecular properties computed from Chemical Component Dictionary SMILES via RDKit (heavy atom count, rotatable bonds, hydrogen-bond acceptors and donors, octanol-water partition coefficient, ligand type classification) (PDB), ligand contacts with pocket secondary-structure composition and pocket flexibility from B-factors (PDB), catalytic activities.
- Domain architecture (5): multi-domain layout, linker regions, domain–domain interfaces, UniProt (The UniProt Consortium, 2025) domain regions remapped to PDB coordinates via SIFTS (Dana et al., 2019) (all PDB); chain layout.
- Function and annotation (5): GO (The Gene Ontology Consortium, 2021) molecular function terms filtered to experimental evidence (PDB), subcellular locations from UniProt comments (PDB), post-translational modifications remapped via SIFTS (PDB), behavior statements (PDB), free-text function.
- PDB-specific annotations (4; PDB only): membrane topology from PDBTM (Kozma et al., 2013), tunnel geometry from ChannelsDB (Pravda et al., 2018), curated repeats from RepeatsDB (Paladin et al., 2021), antibody CDR regions from SAbDab (Dunbar et al., 2014).

B.3 Residue-specific captions

Most captions describe properties without identifying the participating residues, as in `three disulfide bonds`. For 15 properties, captions can also include positions in 1-based SEQRES numbering, as in `disulfide between Cys8 and Cys38`. We sample independently for each property in each caption whether to include these positions. Properties whose individual positions define the requested structure, including metal coordination, catalytic activity, and disulfides, receive positions more often (p=0.35–0.40). Interaction properties such as interface hydrogen bonds, salt bridges, and hydrophobic contacts use lower probabilities (p=0.10–0.20). We omit positions for emergent properties such as flexibility, hinge residues, and β-turns because individual residues are not independently designable in these cases.

Residue identifiers use standard structural biology notation, such as `Cys8` and `Arg118`. Cross-chain interactions add a `target:` prefix to partner-chain residues, as in `Arg45-target:Asp112`. Captions that include positions omit qualitative count terms such as `several` and `many`, and long position lists are truncated to a fixed number of entries. The training distribution therefore mixes exact positional prompts with position-free descriptions.

B.4 Caption sampling

Including every available property in every caption would make the descriptions repetitive and dense. We instead generate multiple captions per structure and vary the number of properties included in each one. For a structure with n available properties, we sample a continuous proportion $X \sim \mathrm{Beta}(1, \beta)$ and set

$$K = \lfloor nX \rfloor + 1,$$

where K is the number of properties included in the caption. From the available set, we sample K properties uniformly without replacement. Setting $\beta = 5.0$ favors sparse captions while retaining occasional dense descriptions.

The expected number of sampled properties is

$$\mathbb{E}[K] = \sum_{i=1}^{n} P(K \geq i) = \sum_{i=1}^{n} P\big(X \geq \tfrac{i-1}{n}\big) = \sum_{i=1}^{n} \big(1 - \tfrac{i-1}{n}\big)^{\beta}, \tag{3}$$

where the last equality uses the survival function of $\mathrm{Beta}(1, \beta)$. We choose the number of captions per structure by bounding the probability that any property is never sampled. Let $q = 1 - \mathbb{E}[K]/n$ approximate the probability that one caption excludes a property, and let E_i denote the event that property i is absent from all N captions. The union bound gives

$$P(E_1 \cup \cdots \cup E_n) \leq nq^N. \tag{4}$$

Setting this bound to $1 - p$ gives $N = \lceil (\log(1-p) - \log n)/\log q \rceil$, with $p = 0.95$. The number of populated properties varies among entries because, for example, many PDB chains lack ligands or metal-coordination sites. Applying the coverage criterion produces averages of approximately 32 captions per PDB entry, 26 per AFDB entry, and 28 per Teddymer chain.

B.5 LLM CAPTION GENERATION

We generate captions with Qwen3-235B-A22B (Yang et al., 2025) served through vLLM. The input is a structured list of sampled annotations, and the prompt instructs the model to produce a faithful caption without adding information beyond them (Box B.1). We sample with temperature 0.7, top-p 0.8, top-k 20, and presence penalty 1.5. Caption generation runs locally. The measured throughput of approximately 14,300 captions per H200 GPU-hour places the cost of generating all 52.4M captions at approximately 3,700 H200 GPU-hours.

Caption generation prompt and example output

```
You are given a list of facts about a protein formatted as
key-value pairs.  Generate a factual protein caption from the
following facts.  Only state what is explicitly provided.  Do not
speculate, infer, or elaborate beyond the given information.

Provide only the caption text.

Facts:
- Structural class:  Alpha Beta
- Helices:  4
- Strands:  5
- Oligomeric context:  monomeric (C1)
- Repeat structure:  no internal repeats detected

Caption:
The protein adopts an alpha-beta structural class, featuring
a total of four helices and five strands.  It functions as a
monomeric entity in its oligomeric context.
```

Validation. We evaluate caption fidelity with Claude Opus 4.6 as an LLM judge on 80 captions. The sample covers different properties, data sources, caption lengths, and the presence or absence of residue-specific detail. For each caption, the judge compares every factual claim against the source annotations and evaluates three metrics:

- Recall: fraction of input properties whose content appears in the generated caption.
- Precision: fraction of distinct factual claims in the caption that are traceable to a source annotation. Mild editorial language (e.g., "contributing to structural stability") counts as traceable.
- Accuracy: fraction of mentioned properties that are stated with the correct value.

Table 5: **Caption validation.** The evaluation covers 80 captions.

Metric	Claims	%
Recall	584 / 597	97.8
Precision	804 / 895	89.8
Accuracy	774 / 809	95.7

Unsupported numerical specificity accounts for most of the caption judge's precision loss. Qualitative annotations such as `high` shape complementarity or `extensive` buried surface area sometimes prompt the language model to add a plausible value such as `score 0.89` or `3700 Å²`. A less frequent failure divides a total into unsupported components, such as surface and buried contributions to a salt-bridge count. Despite these errors, the captions retain high recall, precision, and accuracy on the audited sample (Table 5).

B.6 TEXT–RESIDUE ALIGNMENT ANNOTATIONS

After generating captions, we map text spans to residue positions to produce supervised alignment masks. The selected source annotations identify which property text to locate in each caption. We find that text with regular-expression anchors and TF–IDF-weighted token matching against the formatted annotation. Captions that include residue positions use direct matches to identifiers such as `Cys8` (§B.3). For example, given the caption in Box B.1 and the annotation `Structural class: Alpha Beta`, the annotator computes TF-IDF weights for the tokens `structural`, `class`, `alpha`, and `beta`, scans the caption, and identifies `alpha-beta structural class` as the best-matching region. We then map this span to the helix and strand residues within the CATH domain boundary.

The property type determines the residue set assigned to each matched text span. Secondary-structure phrases map to helix, strand, or sheet residues, and CATH fold labels map to secondary-structure residues within the annotated domain boundary. Disulfides, salt bridges, metal coordination, and interface interactions map to both participating residues as pairs. Whole-chain properties such as contact order and compactness map to all modelled positions. Database-level properties, including ECOD labels, quaternary structure, Gene Ontology terms, subcellular locations, and free-text function, receive no residue mask. We omit an annotation from the mask when its text span cannot be located reliably.

Validation. We manually label the character span corresponding to each selected property in 80 captions, split evenly between held-in and hold-out examples, for 617 spans in total. The annotator finds spans for nearly every category in the 40 hold-out captions, with lower recall where connecting words fall between matched tokens (99% coverage, 89% character-level precision, and 73% recall over 267 spans). Examples of missed connecting text include `corresponding to` and `with a mean distance of`.

C MODEL AND TRAINING DETAILS

Prompteina adds text conditioning to Complexa while retaining its partially latent flow-matching representation. This appendix specifies the inherited notation, new conditioning components, training, and sampling; Didi et al. (2026) describe the full base architecture.

C.1 INHERITED COMPLEXA MODEL

We initialize Prompteina from a Complexa checkpoint trained first on AFDB monomers and then on PDB and Teddymer complexes. Complexa uses partially latent flow matching to generate backbone C_α coordinates $\mathbf{x}^{\mathrm{C}_\alpha} \in \mathbb{R}^{N\times 3}$ and per-residue side-chain latents $\mathbf{z} \in \mathbb{R}^{N\times 8}$. A frozen variational autoencoder $\mathcal{E}$ maps an atomistic structure $\mathbf{x}$ to these latents. For a training structure $(\mathbf{x}_1^{\mathrm{C}_\alpha}, \mathbf{z}_1) \sim p_{\mathrm{data}}$ with $\mathbf{z}_1 = \mathcal{E}(\mathbf{x})$, the model samples independent timesteps $t_x, t_z \in [0,1]$. It constructs linear interpolants $\mathbf{x}_{t_x}^{\mathrm{C}_\alpha} = t_x\mathbf{x}_1^{\mathrm{C}_\alpha} + (1-t_x)\mathbf{x}_0^{\mathrm{C}_\alpha}$ and $\mathbf{z}_{t_z} = t_z\mathbf{z}_1 + (1-t_z)\mathbf{z}_0$ from noise $\mathbf{x}_0^{\mathrm{C}_\alpha} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\mathbf{z}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The denoiser $\mathbf{v}_\phi$ predicts the velocity field:

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{t_x,t_z}\left[\left\|\mathbf{v}_\phi^x - (\mathbf{x}_1^{\mathrm{C}_\alpha} - \mathbf{x}_0^{\mathrm{C}_\alpha})\right\|^2 + \left\|\mathbf{v}_\phi^z - (\mathbf{z}_1 - \mathbf{z}_0)\right\|^2\right]. \tag{5}$$

Here $\mathbf{v}_\phi^x$ and $\mathbf{v}_\phi^z$ are evaluated at $(\mathbf{x}_{t_x}^{\mathrm{C}_\alpha}, \mathbf{z}_{t_z}, \mathbf{c}, t_x, t_z)$.

The denoiser is a transformer with hidden width d over N residue states $\mathbf{h} \in \mathbb{R}^{N\times d}$. Each layer applies adaptive layer normalization (AdaLN), pair-biased multi-head self-attention, and a feed-forward network:

$$\begin{aligned}\mathbf{h} &\leftarrow \mathbf{h} + \mathrm{AdaScale}\big(\mathrm{MHSA}\big(\mathrm{AdaLN}(\mathbf{h}, \mathbf{c})\big), \mathbf{c}\big),\\ \mathbf{h} &\leftarrow \mathbf{h} + \mathrm{AdaScale}\big(\mathrm{FFN}\big(\mathrm{AdaLN}(\mathbf{h}, \mathbf{c})\big), \mathbf{c}\big).\end{aligned} \tag{6}$$

Let d_c denote the width of the conditioning array $\mathbf{c} \in \mathbb{R}^{N\times d_c}$. In Complexa, $\mathbf{c}$ contains only embeddings of t_x and t_z. Prompteina also includes the pooled text representation in $\mathbf{c}$.

C.2 TEXT ENCODERS

Sentence encoder. We use Qwen3-Embedding-8B through the `sentence_transformers` library to produce L2-normalized, 4,096-dimensional embeddings. Each caption receives the following task instruction before encoding:

Sentence encoder task instruction

```
Given a description of a protein, embed it based on semantic
meaning only.  Ignore surface form variation such as casing,
punctuation, paraphrasing or rewording.
```

The instruction asks the frozen encoder to ignore stylistic variation in the generated captions. The two-layer MLP g_s projects the embedding with Linear $4096\to1024$, GELU, LayerNorm, and Linear $1024\to256$.

Token encoder. We use Qwen3-8B with its default chat-template formatting. We extract the hidden state from each of its 36 transformer layers, excluding the embedding layer, and average them for each token. The two-layer MLP g_t projects each token with Linear $4096\to1024$, GELU, LayerNorm, and Linear $1024\to768$.

We freeze both encoders throughout all training stages.

C.3 TOKEN CROSS-ATTENTION

For a prompt with T tokens, let L denote the number of language-model layers and d_{LM} their hidden width. Let $\mathbf{e}_i^{(l)} \in \mathbb{R}^{d_{\mathrm{LM}}}$ denote the representation of token i at language-model layer l. We aggregate the L representations for each token with a function f:

$$\tilde{\mathbf{c}}_i^{\mathrm{t}} = f\big(\mathbf{e}_i^{(1)}, \ldots, \mathbf{e}_i^{(L)}\big) \in \mathbb{R}^{d_{\mathrm{LM}}}, \quad i = 1, \ldots, T.$$

The best-performing choice in the ablation is an average across layers (App. D). The learned projector gives $\mathbf{c}_i^{\mathrm{t}} = g_t(\tilde{\mathbf{c}}_i^{\mathrm{t}}) \in \mathbb{R}^d$.

The function $\mathrm{CrossAttn}(\mathbf{h}, \mathbf{c}^{\mathrm{t}}, \mathbf{c})$ denotes cross-attention from the protein states $\mathbf{h}$ to the token embeddings $\mathbf{c}^{\mathrm{t}}$, with $\mathbf{c}$ modulating the protein queries. Let α_ℓ be the learned scalar gate for denoiser layer ℓ. Each layer applies the following update before protein self-attention:

$$\mathbf{h} \leftarrow \mathbf{h} + \tanh(\alpha_\ell)\,\mathrm{CrossAttn}(\mathbf{h}, \mathbf{c}^{\mathrm{t}}, \mathbf{c}). \tag{7}$$

Let d_h denote the attention-head width. Let LN denote layer normalization, and let $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V$ denote the learned query, key, and value projections. The projected queries, keys, and values are $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$. Let Attn denote scaled dot-product attention. Protein states form the queries, and text tokens form the keys and values:

$$\mathrm{CrossAttn}(\mathbf{h}, \mathbf{c}^{\mathrm{t}}, \mathbf{c}) = \mathrm{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}),$$
$$\mathbf{Q} = \mathrm{AdaLN}(\mathbf{h}, \mathbf{c})\mathbf{W}_Q \in \mathbb{R}^{N\times d_h}, \quad \mathbf{K} = \mathrm{LN}(\mathbf{c}^{\mathrm{t}})\mathbf{W}_K \in \mathbb{R}^{T\times d_h}, \quad \mathbf{V} = \mathrm{LN}(\mathbf{c}^{\mathrm{t}})\mathbf{W}_V \in \mathbb{R}^{T\times d_h}. \tag{8}$$

An output projection $\mathbf{W}_O$ maps the attention output back to dimension d. We initialize α_ℓ to zero. The conditioning array $\mathbf{c}$ enters through AdaLN on the queries, so the pooled text representation and denoising timestep modulate the token–protein interaction.

C.4 Text–residue alignment head

Let $\Delta\mathbf{h}_\ell = \tanh(\alpha_\ell)\mathrm{CrossAttn}(\mathbf{h}, \mathbf{c}^{\mathrm{t}}, \mathbf{c}) \in \mathbb{R}^{N\times d}$ denote the gated cross-attention update at denoiser layer ℓ. Let L_d denote the number of denoiser layers. We normalize learned weight logits $\mathbf{w}$ as $\boldsymbol{\beta} = \mathrm{softmax}(\mathbf{w})$ and aggregate the updates as $\bar{\mathbf{h}}$:

$$\bar{\mathbf{h}} = \textstyle\sum_\ell \beta_\ell\, \Delta\mathbf{h}_\ell \in \mathbb{R}^{N\times d}.$$

A zero-initialized projection $\mathbf{W}_a \in \mathbb{R}^{d\times d}$ maps $\bar{\mathbf{h}}$ into a shared space with the token embeddings. Their dot product gives the alignment logits:

$$A_{n,i} = \langle \mathbf{W}_a \bar{\mathbf{h}}_n, \mathbf{c}_i^{\mathrm{t}} \rangle.$$

We train these logits with Equation 1. Aligned residue–token pairs are sparse, so positive entries are upweighted. Let n_{pos} and n_{neg} denote the number of positive and negative entries in one sample. Samples without valid alignment annotations receive zero alignment loss. Otherwise, the per-sample positive weight is $w_+ = n_{\mathrm{neg}}/n_{\mathrm{pos}}$.

C.5 Initialization and gating

AdaLN and AdaScale. The base model's AdaLN computes $\gamma(\mathbf{c}) \odot \mathrm{LayerNorm}(\mathbf{h}) + \beta(\mathbf{c})$, where $\gamma = \sigma(\mathbf{W}_\gamma \mathbf{c} + \mathbf{b}_\gamma)$ and $\beta = \mathbf{W}_\beta \mathbf{c}$, with default initialization. AdaScale computes $\sigma(\mathbf{W}_s \mathbf{c} + \mathbf{b}_s) \odot \mathbf{h}$. Its weights start at zero and its bias at -2.0, so $\sigma(-2.0) \approx 0.12$ and each layer's initial output contribution is small.

Cross-attention gating. We gate the cross-attention output with a per-layer scalar α_ℓ initialized to zero:

$$\Delta\mathbf{h}_\ell = \tanh(\alpha_\ell) \cdot \mathrm{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}).$$

The zero initialization removes the cross-attention contribution at the start of training and allows it to grow thereafter. The ablation compares this gate with AdaScale gating, AdaScale combined with the tanh gate, unbounded linear conditioning, per-dimension learned scalars, and no gating (App. D; Peebles & Xie, 2023).

Alignment head projection. We zero-initialize $\mathbf{W}_a \in \mathbb{R}^{d\times d}$ so that the alignment head produces zero logits at the start of training, allowing the generation objective to stabilize before alignment gradients take effect.

C.6 Training

Training first introduces the text pathways for monomer generation and then adapts the model to generate a binder against a fixed target structure. The monomer phase contains three stages.

Monomer training. We follow the staged adaptation used by PixArt-α (Chen et al., 2023). The first stage warms up the text projectors g_s and g_t, cross-attention parameters $\{\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V, \mathbf{W}_O, \alpha_\ell\}$, and AdaLN conditioning layers on paired text–structure data while freezing the pretrained denoiser $\mathbf{v}_\phi$. The second stage unfreezes all parameters for text-conditioned continual pretraining on PDB and AFDB monomers with $\mathcal{L}_{\mathrm{FM}}$. The third stage removes the flow-matching loss and fine-tunes the alignment pathway with $\mathcal{L}_{\mathrm{align}}$. During this stage, we freeze the denoiser, AdaLN layers, cross-attention gates, and query and key projections. We train the value and output projections in the last five cross-attention layers together with the text projectors and alignment head $\mathbf{W}_a$.

Restricting alignment fine-tuning to the late value and output projections preserves fold accuracy. Higher learning rates or unfrozen query and key projections degrade fold accuracy within one to two epochs, whereas training the last five cross-attention layers matches all fourteen layers with one third as many trainable parameters. During warmup and both continual-pretraining stages, we replace $\mathbf{c}^{\mathrm{s}}$ and $\mathbf{c}^{\mathrm{t}}$ with zero vectors with probability $p_{\mathrm{uncond}} = 0.2$. This dropout trains the unconditional prediction used for classifier-free guidance.

Binder training. We continue training on Teddymer dimers (Didi et al., 2026) and PDB multimers. Only the binder tokens are noised, leaving the target tokens clean and fixed for the denoiser to attend to at every step. An optional hotspot mask marks target residues that the binder should contact and enters through the pair and sequence features used for the target. We sample hotspots from true contact residues with dropout so that the training distribution also includes prompts without a specified epitope. Input text describes the binder, while the target structure enters separately.

Training the final monomer and binder models required approximately 12,900 H100-equivalent GPU-hours, or about 67 days on one eight-H100 node. This estimate excludes ablations, dataset construction, and evaluation.

Every training stage uses a 750-step linear warmup, an effective batch size of 128, and exponential-moving-average decay of 0.999 (Table 6). The learning rate is 1.8×10^{-4}, except for alignment fine-tuning at 5.4×10^{-6}.

Table 6: **Training stages.** "Trainable" identifies components with gradients enabled; all other components are frozen. Steps are optimizer steps at the selected checkpoint.

Stage	Dataset	Trainable	Loss	LR	Steps
Projector warmup	PDB+AFDB monomers	Text modules, AdaLN	$\mathcal{L}_{\mathrm{FM}}$	1.8×10^{-4}	2.6K
Continual pretraining (monomers)	PDB+AFDB monomers	All	$\mathcal{L}_{\mathrm{FM}}$	1.8×10^{-4}	816K
Alignment fine-tuning	PDB+AFDB monomers	Cross-attn $\mathbf{W}_V$/out (late layers), projectors, $\mathbf{W}_a$	$\mathcal{L}_{\mathrm{align}}$	5.4×10^{-6}	10.2K
Continual pretraining (binders)	Teddymer+PDB dimers	All	$\mathcal{L}_{\mathrm{FM}}$	1.8×10^{-4}	465K

Training distribution rebalancing. A small number of architectures account for most PDB chains, so uniform chain sampling presents common folds many times per epoch and rare folds only a few times. We cluster structures by the structural features described in their captions and interpolate each cluster's sampling weight between its empirical frequency and a uniform distribution over clusters. A parameter $\beta \in [0, 1]$ controls this interpolation. Setting β=0 recovers uniform sampling over chains, and larger values progressively upweight rare feature combinations, by roughly an order of magnitude for the rarest architectures. A cosine schedule alternates between the natural distribution and phases that emphasize rare folds. Rebalancing produces the largest measured gain in fold control, and cosine cycling performs comparably to a monotonic ramp (CATH class accuracy 92.0% with cosine cycling, 87.3% with a linear ramp, and 51.3% without rebalancing; App. D).

Timestep distribution. Complexa samples timesteps from $p_{t_x} = 0.02 \cdot \mathrm{Unif}(0, 1) + 0.98 \cdot \mathrm{Beta}(1.9, 1.0)$ and $p_{t_z} = 0.02 \cdot \mathrm{Unif}(0, 1) + 0.98 \cdot \mathrm{Beta}(1.0, 1.5)$, concentrating training near clean data ($t \to 1$). We replace the backbone schedule with $p_{t_x} = \mathrm{Unif}(0, 1)$ and retain the side-chain

schedule p_{t_z}. The uniform backbone schedule gives more weight to intermediate timesteps, where the model makes structural decisions and text conditioning is most informative.

Alignment head representation. The alignment head uses cross-attention updates as its source representation. The representation ablation compares these updates with trunk hidden states and output-head activations (App. D).

C.7 SAMPLING

At inference, Prompteina generates backbone coordinates and side-chain latents by simulating a stochastic differential equation (SDE). The frozen variational autoencoder (VAE) decoder $\mathcal{D}(\mathbf{x}^{C_\alpha}, \mathbf{z})$ reconstructs all-atom coordinates and amino acid sequence. Sampling follows Geffner et al. (2025); Didi et al. (2026) for 400 Euler–Maruyama steps.

Classifier-free guidance (Ho & Salimans, 2022) combines conditional and unconditional velocities:

$$\mathbf{v}_{\text{guided}} = w\,\mathbf{v}_{\text{cond}} + (1-w)\,\mathbf{v}_{\text{uncond}}, \tag{9}$$

where w controls the strength of text conditioning. The evaluation protocol specifies the guidance weights and length distributions used for each experiment (App. E).

D ABLATIONS

We ablate the three choices that determine how text enters the model and which training examples and timesteps supply its learning signal: the conditioning architecture, the β data-sampling schedule, and the noise schedule. All ablations share the same parent checkpoint, dataset, and training duration of 20 epochs. Evaluation uses the same protocol as §3.1: 50 structures per prompt, 318 prompts, and CFG w=2.0.

Variants. *Complete model*: pooled + cross-attention, cosine β cycling, uniform noise. *Cross-attn only*: removes the pooled/AdaLN text signal. *Pooled only*: removes cross-attention and the token-level encoder. *No β*: uniform cluster sampling (β=0), removing rare-class rebalancing. *Linear β*: monotonic β ramp 0→1 (vs. cosine cycling). *Default noise*: backbone Cα noise sampled from Beta(1.9, 1.0) instead of uniform.

Ablation results. The complete model obtains 92.0% CATH class accuracy and 27.0% CATH architecture accuracy (Table 7). CATH class accuracy is 86.7% with cross-attention alone, 68.7% with the pooled pathway alone, and 92.0% with both. It is 51.3% without rare-class rebalancing, 87.3% with a linear β schedule, and 92.0% with cosine β cycling. Uniform and inherited Beta backbone-noise schedules give 92.0% and 68.7% CATH class accuracy, respectively.

Table 7: **Architecture ablations.** All metrics use CFG $w = 2.0$. Categorical properties report accuracy, and count properties report Spearman ρ. Rows match Table 10 where applicable. **Bold** and underline denote the best and second-best values.

Group	Property	Complete	Linear β	No β	Cross-attn only	Default noise	Pooled only
Fold & topology, accuracy (%) (↑)							
	CATH class	**92.0**	87.3	51.3	86.7	68.7	68.7
	CATH architecture	**27.0**	25.9	22.2	25.0	18.7	7.5
	Contact order	29.6	**36.4**	28.8	28.4	31.2	33.2
	Internal symmetry	16.0	17.6	16.9	**22.2**	14.9	10.4
	Sheet topology	48.0	50.0	**70.0**	**70.0**	66.0	62.0
	Domain class comp.	**4.7**	4.6	3.3	2.1	0.7	1.6
	Domain arch. comp.	**13.3**	10.5	11.3	10.5	6.3	0.5
Secondary structure, ρ (↑)							
	Helix count	0.87	0.86	0.87	0.87	**0.88**	0.59
	Strand count	0.75	0.70	**0.76**	0.74	0.72	0.58
	Sheet count	0.52	0.60	**0.62**	0.59	0.57	0.44
Tertiary structure, ρ (↑)							
	Disulfide count	0.16	0.21	0.04	−0.11	**0.24**	0.08
	Salt bridge count	0.14	0.20	**0.29**	0.03	0.01	0.10
Global shape & surface, accuracy (%) (↑)							
	Compactness (size)	**45.3**	43.3	36.7	38.0	39.3	39.3
	Compactness (shape)	56.0	**72.0**	60.7	58.7	36.7	45.3
	Surface planarity	**63.0**	53.0	61.0	57.0	44.0	52.0
	SAP	**38.7**	36.0	34.7	34.0	36.7	35.3
Additional metrics							
	CATH topology (%)	**1.6**	1.1	1.1	1.5	1.1	0.4
	Buried unsatisfied (ρ)	0.31	**0.40**	0.36	0.25	0.33	0.29
	Flexibility (ρ)	0.43	0.45	0.43	0.46	0.45	**0.46**

Natural-language and CATH-label comparison. We compare natural language with a dedicated hierarchical CATH-label conditioner using the same structure generation model, training structures, training step, β schedule, requested labels and lengths, evaluator, and CFG. Pooled text and CATH labels perform similarly through AdaLN, whereas adding token cross-attention to the text path substantially raises CATH class and architecture accuracy, and longer CATH-label training to 3.2 times the matched budget does not close this gap (Table 8).

Table 8: **Natural-language and CATH-label conditioning.** The Complete, Pooled only, and 47k CATH-label columns use the same 47,275-step training budget; the final two columns extend only CATH-label training. Values are GearNet top-1 accuracy (%) at CFG $w = 2.0$ over three CATH classes and 35 CATH architectures, with 50 structures per requested label. **Bold** marks the best value in each row.

Property	**Natural language**		**Dedicated CATH label**		
	Complete	Pooled only	47k (matched)	98k	149k
CATH class accuracy (%)	**92.0**	68.7	66.7	73.3	67.3
CATH architecture accuracy (%)	**27.0**	7.5	8.5	9.9	9.7

Alignment fine-tuning comparison. We assess alignment supervision separately from the shared-parent ablations in Table 7. This sequential comparison fine-tunes the text-conditioned monomer checkpoint after 816K optimizer steps for 10.2K additional steps with text–residue alignment supervision. Both checkpoints are evaluated with the same length-matched protocol at CFG w=2.0 and 50 samples per prompt. After alignment fine-tuning, disulfide-count correlation changes from 0.054 to 0.294 and salt-bridge-count correlation from 0.566 to 0.663 (Table 9). CATH class accuracy changes from 94.0% to 96.0%, and architecture accuracy from 55.5% to 51.1%.

Table 9: **Alignment fine-tuning comparison.** The second checkpoint is trained directly from the first, so this sequential comparison is separate from the shared-parent ablations in Table 7. The after-alignment values are those reported for Prompteina in Tables 1 and 10. Both checkpoints use the same length-matched evaluation at CFG $w = 2.0$; disulfide correlation covers requested counts 1–7.

Metric	Before alignment	After alignment
CATH class accuracy (%)	94.0	96.0
CATH architecture accuracy (%)	55.5	51.1
Disulfide count (ρ)	0.054	0.294
Salt bridge count (ρ)	0.566	0.663

E MONOMER GENERATION EVALUATION

We evaluate text-guided monomer generation across 18 tasks: 14 single-property and 4 composite tasks. Each prompt produces 50 structures with classifier-free guidance at w=2.0 and 400 SDE diffusion steps. A prompt requesting property x draws lengths from the empirical training distribution of structures annotated with x, using a piecewise-linear CDF constructed from 11 quantile breakpoints (P5 through P95). For CATH class and architecture, this protocol follows Proteina's treatment of empirical compatibility between chain length and CATH label (Geffner et al., 2025) and avoids unrealistic label–length combinations.

Fold and topology. CATH class (3 prompts) and architecture (35 prompts) are evaluated with GearNet (Zhang et al., 2023), a pre-trained graph neural network for fold classification. We report the fraction of generated structures whose GearNet-predicted CATH label matches the requested label. Contact order (5 prompts) is computed from C_α contacts and binned into 5 classes. We detect nine internal symmetries—cyclic groups C2–C8 and dihedral groups D2–D3—with CE-Symm (Myers-Turnbull et al., 2014).

We evaluate multi-domain CATH class and architecture with separate sets of 16 prompts. Each prompt specifies the class or architecture of every domain in N-to-C order, such as an N-terminal 3-Layer($\alpha\beta\alpha$) Sandwich domain followed by a C-terminal Orthogonal Bundle domain. We segment each generated structure into domains with Chainsaw (Wells et al., 2024) and classify each domain with GearNet. A generated structure is correct only if Chainsaw finds the requested number of domains and GearNet predicts the complete sequence of requested CATH labels in order. All text-conditioned methods receive the same prompts. We restrict the class and architecture compositions to those with at least 500 training examples.

Secondary structure. We compute helix count (20 prompts), strand count (20 prompts), and sheet count (14 prompts) from DSSP assignments (Kabsch & Sander, 1983). For each property, Spearman ρ measures the association between requested and realized counts across all generated structures.

Matched-length control. Property-matched length sampling is necessary because some requests require sufficient chain length; for example, an 80-residue protein cannot reasonably contain 20 helices. For count-valued properties, however, this protocol could also make length predictive of the requested value and thereby inflate requested–realized correlations. We test this potential confound on helix and strand count, for which Prompteina shows strong control in Table 1b. For each property, we select the widest contiguous five-count window whose shared P20–P80 training-length range lies below 200 residues, requiring at least two strands. This gives helix counts 1–5 at 134–193 residues and strand counts 2–6 at 112–169 residues.

Within each property, we reuse the exact same list of 50 lengths at every requested count, making length uninformative about the requested value by construction. We also generate an empty-prompt control using the same lengths. The text-conditioned correlations remain strong, whereas both control correlations are near zero (Table 11).

Tertiary structure. We detect disulfide bonds from S_γ–S_γ distances below 2.5 Å in the generated coordinates and report Spearman ρ over seven requested counts from 1 to 7. For salt bridges, we use Asp/Glu–Lys/Arg atom-pair distances below 4.0 Å and report the correlation across 15 prompts.

Global shape and surface. We evaluate compactness with three prompts each for size and shape. Size is the radius of gyration normalized by chain length, $R_g/N^{0.4}$, and shape is the ratio of the first two singular values of the C_α coordinate matrix. We compute solvent-accessible surface area propensity (SAP, 3 prompts) from the solvent-accessible surface geometry.

Composite evaluations. Four evaluations pair CATH class with a helix, strand, sheet, or disulfide count. Both disulfide-count evaluations use counts 1–7. We report GearNet accuracy for CATH class and Spearman ρ for the count, using the same generated samples for the two reported metrics.

Baselines. Each baseline model is evaluated only on properties its conditioning interface can express. Chroma (Ingraham et al., 2023) text mode covers all categories via free-text caption guidance.

Chroma label mode uses discrete CATH, secondary structure, and radius of gyration conditioners (7 categories). Proteina (Geffner et al., 2025) uses CATH label conditioning with classifier-free guidance (3 categories). RFdiffusion (Watson et al., 2023) uses per-residue secondary structure tensors via scaffoldguided mode (3 categories). RFdiffusion3 (Butcher et al., 2025) does not expose helix-, strand-, or sheet-specific conditioning, and its cyclic or dihedral symmetry sampler generates multi-chain assemblies rather than the single-chain internal symmetry evaluated here.

Pinal (Dai et al., 2025) and BioMatrix (Pei et al., 2026) are text-conditioned models evaluated on the same prompts as Prompteina, and both generate amino acid sequences that we fold with ESMFold. Pinal accepts a target length directly, whereas BioMatrix exposes no length control, so we constrain its decoder to emit exactly the sampled number of residues before terminating. All baselines use their paper-default settings and the same length distributions as Prompteina.

Table 10: **Structural-control benchmark.** We report accuracy (%) for categorical properties and Spearman ρ for count properties. Prompteina uses classifier-free guidance weight w=2.0. Baselines are evaluated only on properties their input format can express (— = not expressible). Best and second-best values are **bold** and underlined, respectively.

	Ours	**Baselines**					
Property	Prompteina	Chroma (text)	Chroma (label)	Proteina	RFD	Pinal	BioMatrix
Fold & topology, accuracy (%) (↑)							
CATH class	96.0	33.3	64.7	**96.7**	—	36.0	30.7
CATH architecture	**51.1**	3.7	7.5	27.5	—	5.5	2.8
Contact order	**56.0**	21.2	—	—	—	27.6	21.6
Internal symmetry	**32.4**	0.4	—	—	—	1.1	0.4
Multi-domain CATH class	**34.4**	5.3	—	—	—	5.4	2.6
Multi-domain CATH arch.	**42.0**	1.5	—	—	—	0.4	1.0
Secondary structure, ρ (↑)							
Helix count	**0.94**	0.74	0.86	—	0.87	0.72	0.59
Strand count	0.88	0.42	0.58	—	**0.92**	0.25	0.18
Sheet count	0.64	0.40	0.59	—	**0.82**	0.16	0.18
Tertiary structure, ρ (↑)							
Disulfide count	**0.29**	0.06	—	—	—	0.10	0.07
Salt bridge count	**0.66**	0.64	—	—	—	0.23	0.28
Global shape & surface, accuracy (%) (↑)							
Compactness (size)	**76.7**	34.0	39.3	—	—	36.7	33.3
Compactness (shape)	**71.3**	31.3	—	—	—	34.7	30.9
SAP	**38.7**	37.3	—	—	—	34.0	34.0
Composite evaluations, class acc. (%) / count ρ (↑)							
Class × helix	**84.1 / 0.86**	31.9 / 0.61	—	—	—	38.6 / 0.10	35.9 / 0.11
Class × strand	**92.1 / 0.65**	32.4 / 0.38	—	—	—	35.5 / 0.21	30.7 / 0.35
Class × sheet	**94.4 / 0.46**	36.7 / 0.34	—	—	—	33.9 / 0.06	28.1 / 0.07
Class × disulf. count	**90.3 / 0.43**	34.5 / −0.02	—	—	—	33.4 / 0.10	31.0 / 0.07

Table 11: **Matched-length secondary-structure control.** Within each property, the same 50-length list is reused at every requested count and for the empty-prompt control ($n = 250$ per row). Control groups inherit the text-conditioned samples' count labels only to compute the null correlation; no count is requested in their prompts. Slope is from a linear regression of realized count on the count label.

Experiment	Requested counts	Lengths (aa)	n	Spearman ρ	Slope
Helix	1–5	134–193	250	**0.878**	**1.076**
Helix—control	—	134–193	250	−0.060	−0.060
Strand	2–6	112–169	250	**0.718**	**0.824**
Strand—control	—	112–169	250	−0.005	−0.010

E.1 PROMPT PARAPHRASE ROBUSTNESS

We test whether control of three well-supported properties depends on the benchmark's exact prompt template: CATH class, helix count, and their composition. For each semantic specification, we compare direct, short-form, paraphrased, and reordered wording (Box E.1). The paraphrases were written with Claude rather than the Qwen model used to generate training captions. All four formats use the same 20 chain lengths for each specification, so wording is the only manipulated variable. All samples are generated specifically for this matched experiment rather than reused from Table 10; there are therefore 20 rather than 50 samples per specification.

Monomer paraphrase examples

```
CATH class: Alpha Beta
Direct: This protein has an Alpha Beta structural class.
Short-form: CATH class: Alpha Beta.
Paraphrased: The overall fold falls into the Alpha Beta category.
Reordered: An Alpha Beta structural class describes this protein.

Helix count: 6
Direct: This protein contains 6 helices.
Short-form: 6 helices.
Paraphrased: The structure is built from six alpha helices.
Reordered: Six helices make up this protein.

Composite: Alpha Beta class and 4 helices
Direct: This protein has an Alpha Beta structural class and
contains 4 helices.
Short-form: Alpha Beta class; 4 helices.
Paraphrased: With four alpha helices, this protein adopts an Alpha
Beta fold.
Reordered: Build a protein with four helices whose overall
structural class is Alpha Beta.
```

Table 12: **Prompt paraphrases.** CATH class uses three specifications, helix count uses five, and the composite uses six; each specification has 20 matched-length generations per format. We report accuracy for CATH class, Spearman ρ for helix count, and accuracy / ρ for their composition.

Evaluation	Direct	Short-form	Paraphrased	Reordered
CATH class accuracy (%)	100.0	100.0	95.0	96.7
Helix count ρ	0.953	0.942	0.912	0.934
Class $\times$ helix (acc. % / ρ)	92.5 / 0.734	87.5 / 0.652	90.8 / 0.729	86.7 / 0.694

All four formats retain strong control of the tested properties (CATH class accuracy at least 95.0%, helix-count ρ at least 0.912, and composite class accuracy and helix-count ρ at least 86.7% and 0.652). Reversing the order of the two composite properties leaves both metrics similar, so control does not depend on which property appears first.

E.2 DESIGNABILITY

Following the standard self-consistency evaluation for generated backbones (Yim et al., 2023; Watson et al., 2023), we sample eight sequences for each structure with ProteinMPNN (Dauparas et al., 2022), fold them with ESMFold, and take the minimum C_α self-consistency RMSD (scRMSD) to the generated structure. This best-of-eight inverse-folding test asks whether at least one designed sequence is predicted to realize the proposed backbone, rather than requiring the generation model's own sequence to do so. Following prior backbone-generation evaluations (Yim et al., 2023; Watson et al., 2023), we classify a structure as designable when scRMSD is below 2 Å for every method.

For every prompt and method, we evaluate an evenly spaced 10-of-50 subset (every fifth structure after sorting by sample index); this preserves coverage of all test cases while controlling the cost

of eight refolds per structure. Each baseline supports a different benchmark subset, so each row of Table 13 restricts Prompteina to the test cases evaluated by that baseline; rows should therefore be read across, not columns down. This avoids treating unsupported tasks as failures or comparing averages over different task mixtures. Additional prompt-matching examples and refolds appear in App. Fig. 6.

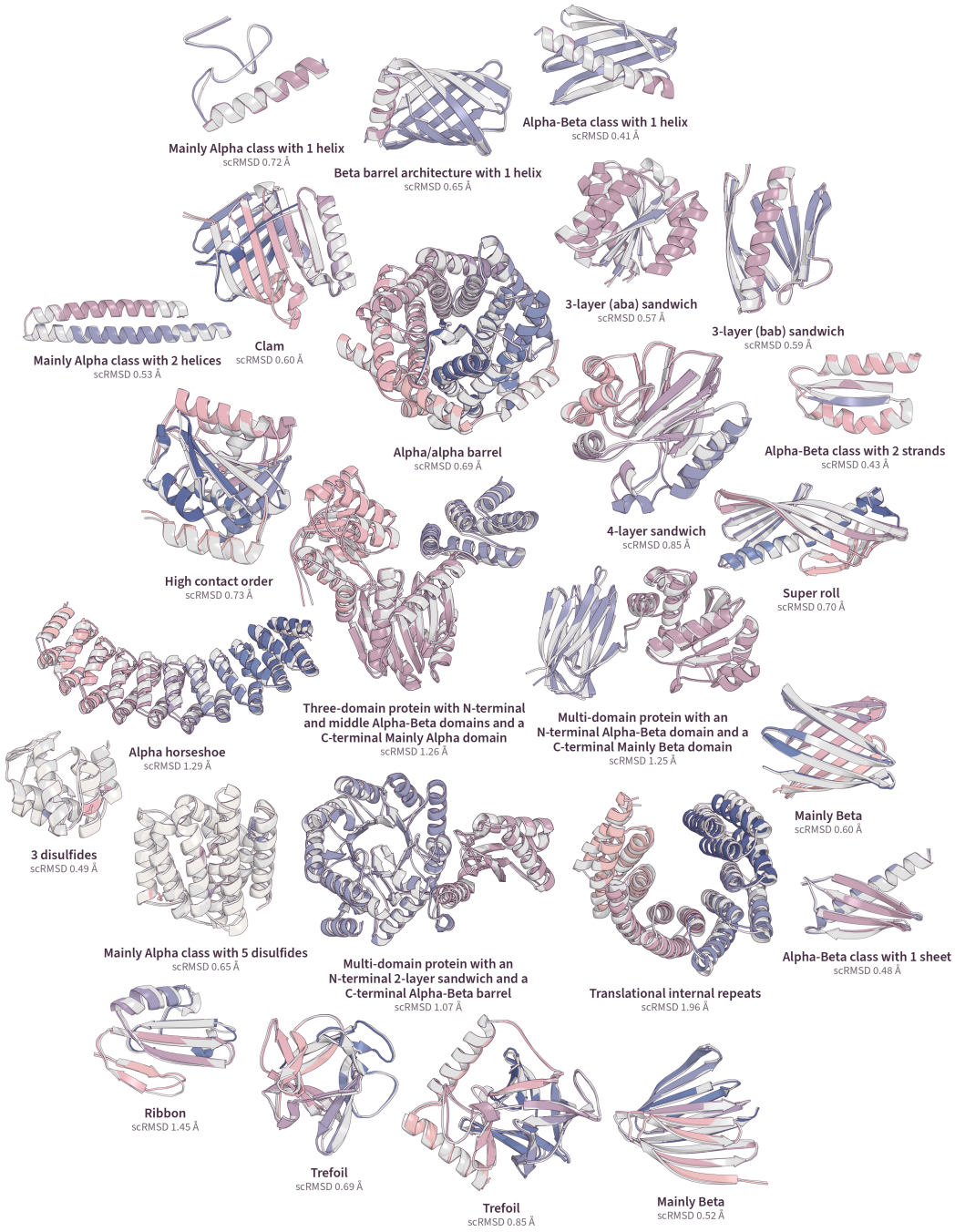


Figure 6: **Additional monomer samples.** Generated structures are colored and superposed with their best of eight ProteinMPNN–ESMFold refolds (light gray) after C_α alignment. Labels give the requested property; values give scRMSD.

Table 13: **Monomer designability.** Each row compares Prompteina with one baseline on exactly the benchmark test cases both evaluated. Median scRMSD is in Å (lower is better); designable is the percentage with scRMSD < 2 Å (higher is better). "Structures per model" gives the sample size on each side of the comparison. Complexa and the text-guided models share all 18 tasks. The property-specific scopes are CATH architecture and class for Proteina, six tasks for Chroma label, and the three secondary-structure-count tasks for RFdiffusion. **Bold** marks the better result within each pair.

Baseline	Shared test cases	Structures per model	Median scRMSD (↓) Prompteina	Median scRMSD (↓) Baseline	Designable (%) (↑) Prompteina	Designable (%) (↑) Baseline
Complexa	251	2,510	**0.83**	3.11	**79.7**	29.1
Text-guided models						
Chroma (text)	251	2,510	**0.83**	14.07	**79.7**	2.4
Pinal	251	2,510	**0.83**	11.10	**79.7**	16.5
BioMatrix	251	2,510	**0.83**	15.83	**79.7**	7.0
Property-specific models						
Proteina	27	270	**0.69**	1.17	**91.1**	64.8
Chroma (label)	84	840	**0.78**	14.80	**81.9**	1.7
RFdiffusion	54	540	**0.83**	1.19	**76.5**	68.1

Conditional and empty-prompt samples use the same fine-tuned checkpoint, the same 251 prompts, and identical per-prompt length distributions; the empty-prompt condition zeros the text embeddings and uses guidance weight $w = 1$. Holding the checkpoint and lengths fixed makes this a control for whether the designability gain comes from fine-tuning or from text conditioning, while preventing the strong dependence of designability on length from confounding the comparison.

Fine-tuned Prompteina is substantially more designable than pretrained Complexa with or without a prompt (79.7% conditional and 85.2% empty-prompt vs. 29.1%). Although the pooled empty-prompt rate is 5.5 percentage points higher, the median paired per-prompt difference is zero; this is not evidence that removing the prompt improves designability. The conditional Prompteina advantage replicates with ESMFold2 (80.7% vs. 26.7% designable; paired scRMSD correlation $\rho = 0.953$). This checks that the conclusion is not specific to one structure predictor; we treat it as a replication with a second folding model, not as a better ground truth.

Prompteina is more designable than all three text-guided full-suite baselines (79.7% vs. 2.4–16.5%). The Proteina comparison uses a different sequence-design instrument because Proteina emits only a C_α trace. The specialist rows also cover narrower task sets.

E.3 Structural novelty

We measure novelty only among designable structures, preventing structurally failed outputs from appearing novel merely because they match no protein fold. For each design, Foldseek (van Kempen et al., 2024) with TM-align searches the training structures used by that model, and novelty is one minus the maximum query-normalized TM-score (Zhang & Skolnick, 2004) to any training structure. Taking the maximum tests the nearest potential training-set match rather than average similarity. Query normalization asks how much of the generated design is explained by that match, so a short design contained within a longer training structure is conservatively treated as non-novel. We use model-specific training references because the relevant memorization opportunity is the corpus to which each model was exposed; consequently, cross-model values are descriptive rather than a controlled comparison against a common database.

Prompteina's conditional and empty-prompt samples share the same 780,195-structure PDB–AFDB training reference, so their matched comparison isolates the effect of supplying text. Both sets begin with 2,510 structures; 2,000 conditional and 2,138 empty-prompt structures pass the scRMSD < 2 Å designability criterion and enter the novelty analysis.

For Complexa, 731 of 2,510 structures are designable. Complexa does not release its original chain-level monomer manifest, so we reconstruct the reported 110,976-chain PDB monomer set from the checkpoint filters and archived PDB metadata, retaining 110,969 after the checkpoint's explicit exclusions. We combine these monomers with the released 45,856-entry PDB multimer manifest.

Expanding multimer entries into their constituent chains and structural models and combining them with the 289,222 AFDB targets gives 506,972 Foldseek targets. Including both PDB components prevents Complexa from appearing artificially novel because part of its training exposure is absent from the search database. Both model-specific references omit Teddymer structures; this can overestimate absolute novelty, but the reconstructed Complexa reference corrects the previous additional omission of its PDB training structures.

Table 14: **Structural novelty.** Each row is searched against the corresponding model-specific PDB–AFDB training reference; reference size counts Foldseek targets. Maximum TM is the best query-normalized structural match to that reference; lower maximum TM and higher novelty indicate greater distance from the model's training set. The two Prompteina conditions use the same checkpoint, prompts, lengths, and reference set. **Bold** marks the better result in this matched conditional–empty-prompt comparison.

Model / condition	Reference size	Max TM (↓) mean / median	Mean novelty (↑)
Complexa	506,972	0.783 / 0.782	0.217
Prompteina (conditional)	780,195	**0.770 / 0.785**	**0.230**
Prompteina (empty prompt)	780,195	0.835 / 0.847	0.165

Conditional Prompteina and Complexa have similar novelty under their model-specific PDB–AFDB references (mean novelty 0.230 vs. 0.217). Text conditioning also moves Prompteina farther from its training structures than the matched empty-prompt control (mean novelty 0.230 vs. 0.165). The matched Prompteina comparison isolates the prompt effect; the cross-model comparison remains descriptive because the reference sets differ and both omit Teddymer.

E.4 Full prompts for Figure 1a

The two prompts abbreviated in Figure 1a are:

- "Mechanobiology sensors need an element that reports tension by stretching reproducibly. It is built as a straight up-down bundle of six alpha helices packed side by side along one long axis, with two beta strands clasping one end. Two disulfide bonds staple the clasp and at least six salt bridges run along the bundle surface. The silhouette is elongated."
- "This enzyme is intended to hydrolyse organophosphate pesticides under EC 3.1.8.1. It adopts a Super Roll architecture of the Alpha Beta class, built from four alpha helices and six beta strands, with the sheet rolled around the helices and a metal centre at the open mouth of the roll. The chain forms zero disulfide bonds and a square silhouette."

F CATH VERIFIER CALIBRATION

Most properties in §3.1 are scored by deterministic procedures, including DSSP hydrogen-bond assignments, CE-Symm symmetry detection, and geometric measurements such as radius of gyration and interatomic distances. Fold classification instead relies on GearNet (Zhang et al., 2023), a learned classifier whose errors affect the measured accuracy. We therefore calibrate GearNet on native PDB structures with known CATH labels. This calibration measures verifier reliability rather than an upper bound on generated-sample accuracy, since generated structures may form a more prototypical subset of a CATH category.

Table 15: **GearNet verifier calibration.** Sample counts are 71,355 for class, 73,163 for architecture, and 71,366 for topology. Weighted accuracy pools structures at each level, whereas the per-category mean gives equal weight to each label.

CATH level	Weighted acc. (%)	Per-category mean (%)	Categories
Class	77.0	79.8	3
Architecture	55.6	44.9	42
Topology	0.2	0.1	1,161

The generation benchmark prompts each category equally, so verifier calibration uses the per-category mean rather than the frequency-weighted accuracy. GearNet is substantially more reliable at the class level than at the architecture level on the prompted categories (79.8% vs. 47.2%). The prompted architecture subset is easier than the full 42-category set because it excludes rare architectures that GearNet handles poorly (47.2% vs. 44.9%). No single CATH class determines the class-level reference (83.5% for Mainly Alpha, 71.5% for Alpha Beta, and 84.4% for Mainly Beta).

GearNet cannot verify topology labels on native structures (0.2% accuracy over 1,161 categories), so we exclude topology from Table 10.

G FUNCTIONAL SPECIFICATION EVALUATION

We test whether a text prompt can specify a protein family and generate recognizable members. The structural properties in §3.1.1 have property-specific verifiers, but no corresponding classifier covers arbitrary protein families.

Benchmark families. The benchmark comprises 122 sequence-coherent protein families. We define each candidate family as an ECOD topology group (Cheng et al., 2014) whose PDB members exceed 30% median pairwise sequence identity, the threshold below which alignment stops implying homology (Rost, 1999). This requirement removes topology groups that share a fold without a single family-level referent.

Eligibility is computed from the median identity over all pairs of PDB sequences in a group before any structure is generated. No design, method, or evaluation score enters the benchmark definition.

Prompt formats. The models were trained with different textual vocabularies, so a single prompt format would favor the model for which that wording is in-distribution. We evaluate every method on four formats and report its highest score for each metric. The *ECOD annotation* combines an ECOD topology name with an EC number or cofactor annotation when available, matching Prompteina's caption vocabulary. The *functional description* paraphrases a Swiss-Prot `FUNCTION` annotation with Claude, matching Pinal's training and evaluation setting. The *protein metadata* combines the protein name with an EC number and subcellular location when available, matching BioMatrix's keyword setting. The *InterPro label* contains only the InterPro entry name, which is a second format used by Pinal (Dai et al., 2025; Pei et al., 2026).

The exact prompts for one benchmark family illustrate these four formats (Box G.1).

Functional-evaluation prompt formats: globin example

ECOD annotation
```
This protein adopts a globin-like fold.  It coordinates a heme
cofactor.
```

Functional description
```
Carries oxygen from the lungs to peripheral tissues.  Its derived
peptide LVV-hemorphin-7 enhances bradykinin activity and thereby
lowers blood pressure, and it also inhibits enkephalin-degrading
enzymes such as DPP3 and selectively antagonizes the pain-signaling
receptor P2RX3, implicating it in the control of pain and
inflammation.
```

Protein metadata
```
hemoglobin subunit beta.
```

InterPro label
```
Globin
```

Generation protocol. Every method generates 50 structures per family and format under the common sampling protocol (App. E). Pinal and BioMatrix emit sequences, which we fold with ESMFold before computing structure similarity.

ProTrek similarity. ProTrek is a contrastively trained trimodal encoder mapping protein sequences, structures, and text into a shared embedding space (Su et al., 2025). Both scores are temperature-scaled cosine similarities between the prompt embedding and a generated sequence or structure embedding.

Confidence-weighted similarity reduces the contribution of generated sequences that ESMFold predicts with low confidence without filtering any design or family. For each design, it multiplies ProTrek similarity by mean per-residue $\mathrm{pLDDT}/100$. We obtain pLDDT by applying ESMFold to the exact generated sequence, using Prompteina's co-designed sequence or the sequence emitted by

Pinal or BioMatrix. Scores are averaged first within each family and then across the 122 families, keeping the family as the unit of analysis.

Table 16: **Functional control across prompt formats.** Mean ProTrek similarity between a prompt and the sequence or folded structure generated from it, over 50 designs for each of 122 protein families. Formats marked † are in-distribution for that method. **Bold** marks each method's best format. ± denotes the half-width of a 95% cluster-bootstrap interval obtained by resampling the 122 families rather than designs within a family.

ProTrek seq. sim. (↑)				
Method	ECOD ann.	Function	Metadata	InterPro
Prompteina	**8.46**† ±0.98	5.96 ±0.91	5.65 ±0.89	5.51 ±0.94
Pinal	**7.55** ±0.79	6.37† ±0.82	6.97 ±0.85	7.52† ±0.72
BioMatrix	4.73 ±0.64	3.49 ±0.56	4.72† ±0.62	**5.73** ±0.68

ProTrek struct. sim. (↑)				
Method	ECOD ann.	Function	Metadata	InterPro
Prompteina	**7.36**† ±1.04	4.33 ±0.91	4.12 ±0.87	4.55 ±0.96
Pinal	6.84 ±0.78	5.27† ±0.84	6.23 ±0.84	**7.09**† ±0.73
BioMatrix	3.57 ±0.56	2.09 ±0.48	3.33† ±0.55	**4.64** ±0.63

Table 17: **Confidence-weighted functional control.** Each design's ProTrek similarity is multiplied by its ESMFold pLDDT divided by 100; no design or family is removed. Each cell reports the highest mean over the four prompt formats. Uncertainty is the half-width of a 95% percentile cluster-bootstrap interval over the 122 families (10,000 replicates).

Method	Seq. sim. (↑)	Struct. sim. (↑)
Prompteina	**7.12** ±0.88	**6.20** ±0.90
Pinal	5.98 ±0.70	5.49 ±0.68
BioMatrix	3.70 ±0.58	3.20 ±0.56

Baseline training and metric overlap. Pinal is designed for text-to-protein generation, whereas BioMatrix is a generalist model instruction-tuned for the same task. Both generate protein sequences from functional descriptions and include UniProt-derived training data. Prompteina is trained to generate all-atom structures from captions that combine structural and functional properties.

ProTrek is not independent of Pinal's training. Pinal's TrEMBL50-ProTrek corpus was built by scoring TrEMBL cluster representatives against Swiss-Prot descriptions with ProTrek and retaining about 412M of the highest-scoring sequence–description pairs (Dai et al., 2025). Pinal's structure tokens use ProTrek's Foldseek tokenization, and Pinal selects candidates by ProTrek score. Prompteina uses neither these data nor these selection procedures.

We use ProTrek to retain the metric reported in the original Pinal and BioMatrix evaluations.

Method comparison. At each method's best prompt format, Prompteina has the highest mean sequence and structure similarity (Table 16). Its unweighted advantage over Pinal is small and unresolved under a paired family bootstrap (sequence $+0.91$ $[-0.14, +1.93]$; structure $+0.26$ $[-0.73, +1.30]$). The corresponding advantages over BioMatrix are larger and resolved (sequence $+2.73$ $[+1.75, +3.68]$; structure $+2.71$ $[+1.74, +3.71]$).

Confidence weighting increases the separation without changing the overall conclusion (Table 17). Prompteina's advantage over Pinal resolves for sequence similarity but remains unresolved for structure similarity (sequence $+1.14$ $[+0.26, +2.03]$; structure $+0.72$ $[-0.15, +1.60]$). Both confidence-weighted advantages over BioMatrix resolve (sequence $+3.42$ $[+2.58, +4.25]$; structure $+3.01$ $[+2.18, +3.85]$).

H BINDER DESIGN EVALUATION

H.1 INTERFACE-RESIDUE STEERING

Interface residue detection. Following Jones & Thornton (1996), we define interface residues as binder residues whose solvent-accessible surface area (SASA) decreases by more than 1.0 Å^2 upon complex formation. SASA is computed with FreeSASA using default radii. The interface residue count reported throughout §3.2.1 is the number of binder-side residues meeting this criterion.

Benchmark targets. We use all 20 targets in the Complexa benchmark (Didi et al., 2026), retaining its target structures, binder-length ranges, and hotspot residues where provided. The agentic binder protocol lists the complete panel below.

Prompt template. We specify only the requested binder–target interface residue count: ``The binding interface involves {n} residues.'' Requested counts range from 8 to 30; scaffold topology, contact chemistry, and other binder properties remain unspecified.

Generation protocol. We sample 50 structures per requested count with classifier-free guidance at $w = 2.0$. The single-pass protocol omits search, reward models, inverse folding, and structure prediction so that these components cannot alter the requested-count response. Within a target, every requested count uses the same 50 binder lengths, sampled uniformly from the target-specific range with a fixed seed. The target structure and benchmark hotspots also remain fixed, leaving the requested contact-residue count as the only manipulated variable.

Unconditional baseline. We sample pretrained Complexa with unchanged target inputs, disabled text, and the same binder lengths as the text-conditioned samples. Classifier-free guidance is undefined when both text and fold conditioning are absent.

Prompt paraphrases. We test interface-residue control under key–value, paraphrased, and reordered prompts on CD45, IL-17A, and PDL1 (Box H.1). For each target and format, we request 8–37 interface residues and generate 20 structures at each of the 30 levels. Every format uses the same 20 binder lengths at a given target and level. Across the 30 requested levels, we report Spearman ρ between requested and mean realized counts and the slope of mean realized on requested count (Table 18). Response magnitude differs by target, so we express each slope relative to that target's key–value slope rather than pooling targets.

Interface-residue paraphrase examples at a requested count of 22

Key--value
```
- Protein-protein interface:  22 interface residues
```

Paraphrased
```
The binding interface involves 22 residues.
```

Reordered
```
22 residues make up the binding interface.
```

All tested phrasings preserve the requested-count ordering ($\rho \geq 0.976$). Paraphrased prompts retain or increase the key–value response slope on every target, while reordered prompts retain 76–109% of the key–value response.

H.2 AGENTIC BINDER DESIGN PROTOCOL

H.2.1 TARGET PANEL

The panel contains 20 protein targets drawn from the AlphaProteo (Zambaldi et al., 2024) and BindCraft (Pacesa et al., 2025) targets aggregated by Didi et al. (2026). We exclude influenza H1 and the SARS-CoV-2 receptor-binding domain because our prompt-expansion pipeline returned no usable

Table 18: **Interface-residue control across prompt formats.** Each row contains 600 structures (30 requested counts × 20 matched lengths). Relative slope divides by the key–value slope separately within each target.

Target	Format	Slope	Relative slope	Spearman ρ
CD45	Key–value	0.851	1.00	0.989
	Paraphrased	0.910	1.07	0.997
	Reordered	0.647	0.76	0.976
IL-17A	Key–value	0.808	1.00	0.997
	Paraphrased	0.933	1.15	0.996
	Reordered	0.881	1.09	0.997
PDL1	Key–value	0.346	1.00	0.985
	Paraphrased	0.437	1.26	0.993
	Reordered	0.366	1.06	0.989

specifications, and apply this exclusion to every method. IL-17A and TNF-α remain in the panel, although Complexa excludes them after no evaluated method succeeds within its runtime budget.

Each target contributes a structure and binder-length range; curated hotspots are available for 18 targets. In the agentic setting, each method receives the structure and length range and selects an epitope for each design. In the non-agentic setting, each method receives those curated hotspots.

H.2.2 Generation protocol

Every method generates 200 designs per target in both the agentic and non-agentic settings. We divide the target's binder-length range into 20 contiguous bands and draw 10 designs in each, preventing a method's length prior from concentrating its samples in part of the range. Designs are sampled independently without inference-time search, a reward model, or candidate selection. The common budget and generation procedure isolate differences among conditioning channels.

Prompteina uses 400 SDE denoising steps as in the monomer evaluation, with classifier-free guidance reduced to $w = 1.0$ (App. E). Baselines use their published inference defaults.

H.2.3 Sequence conditions

Each generated backbone is scored under three sequence conditions. The *self-sequence* condition uses the sequence produced by the structure generation model. The *redesign* condition samples eight ProteinMPNN sequences per backbone and retains the highest-scoring one (Dauparas et al., 2022). The *fixed-interface redesign* condition also uses ProteinMPNN but fixes the generated identities of binder interface residues, changing only the scaffold sequence. A binder residue belongs to the interface when any of its heavy atoms lies within 6 Å of a target heavy atom.

The gap between the two redesign conditions isolates how much of a method's score is carried by ProteinMPNN rewriting the interface rather than by the interface the model generated. Didi et al. (2026) report the same condition for this reason.

Reporting all three conditions permits comparison among methods that use inverse folding differently. RFdiffusion3 co-designs sequence and structure, but its alanine-biased sequence head is followed by ProteinMPNN in every experiment reported by its authors. Scoring only its self-sequences would omit part of the published method; Complexa and Prompteina require no redesign.

H.2.4 Agentic specification generation

We use Claude Opus 4.6 as the agent in a two-stage pipeline. A research stage gives a language model tool access to web and scientific-literature search, PDB structure search and entry lookup, and UniProt lookup. The model uses these sources to prepare a structured dossier about target structure, binding biology, known binders, and biophysical properties (Box H.2). An expansion stage reads the dossier and emits the design specifications consumed by each structure generation model (Box H.3).

Each method receives the same four classes of input, with only the method reference differing:

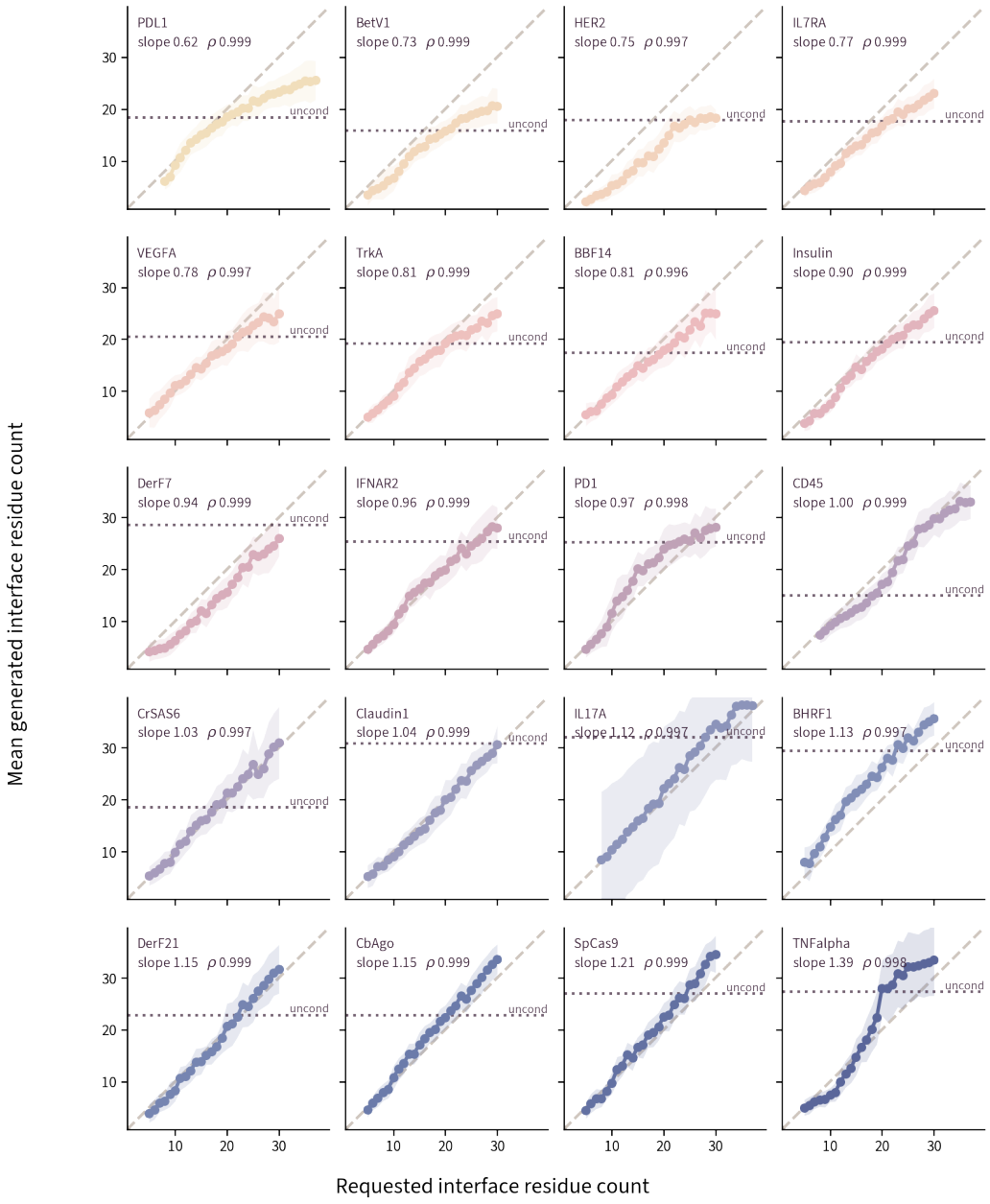


Figure 7: **Interface-residue steering across targets.** Each panel shows the mean realized count across 50 binders for requested counts 8–30; colored bands span one standard deviation. Dashed diagonals mark equality. Dotted horizontal lines mark the per-target unconditional mean. Annotations give the fitted slope and Spearman's ρ; panels are ordered by slope.

- **The research dossier**, byte-identical across methods for a given target.
- **The task instruction**, which states how many designs to produce, the length band, and the requirement that designs be diverse. It is byte-identical across methods and names none of them.
- **A method reference**, the only input that differs. It describes what that method is, the channels it exposes, the closed set of values each channel accepts, and the exact emission format. Each is self-contained and refers to no other method.
- **Structural query tools** over the target, listed in Table 19, together with the target configuration (target chain and residue range, binder length range, PDB identifier).

Table 19: **Structural query tools.** The tools return target properties rather than raw coordinates, which are too large for the language-model context.

Tool	Returns
`target_overview()`	Chains, residue ranges, sequence, length, PDB identifier
`secondary_structure()`	DSSP assignment per chain
`surface_residues(min_sasa)`	Exposed residues with per-residue SASA
`neighbors(residue, radius)`	Residues within r Å, for assembling a contiguous site
`surface_patches(kind)`	Contiguous hydrophobic, polar, or charged patches
`known_interfaces()`	Contact residues from deposited complexes for this target

The `known_interfaces()` tool exposes information already present in the research dossiers and gives every method identical access to deposited complexes. Removing it only from the structural tools would leave the same information available in prose and would not create a stricter comparison.

The agent selects a binding epitope for each design and writes it through the method's available channel: `hotspot_residues` for Prompteina, `select_hotspots` atom pairs for RFdiffusion3, and `binding` properties for BoltzGen. We validate every index against the target's residue numbering. A specification that names an absent residue is rejected and recorded, since replacing it with a nearby residue would change the site selected by the agent.

Apart from clamping proposed binder lengths into the target's range, agent output is not modified before it reaches the structure generation model.

Research stage system prompt (abbreviated)

```
You are a protein biology research assistant with access to search
tools.

Your goal is to help design a STRONG protein binder against the
target.  Build a factual research dossier that will inform binder
design.

PHASE 1 --- IDENTIFY THE TARGET
Call pdb_lookup and uniprot_lookup to identify the target.

PHASE 2 --- GATHER INFORMATION
Search systematically:  find co-crystal structures with binders,
characterize the target surface (shape, epitopes, chemistry), find
known binders (scaffold type, size, BSA, affinity, PDB ID).

PHASE 3 --- SYNTHESIZE (only from tool results)
Write a structured dossier:  (1) Target structure, (2) Binding
biology, (3) Successful binders, (4) Biophysics.

Every PDB ID must be verified by a pdb_lookup call.  Specific
numbers must come from tool results.  Accuracy > completeness.
```

Expansion stage task instruction (abbreviated; identical across methods)

```
A generative model will design protein binders against the target
described in the research dossier.  Using the method reference
provided, produce 10 design specifications.

Every design must have a binder length inside the given band.  Use
the structure tools to choose a binding site on the target, and
name that site explicitly in each design.

The 10 designs must be diverse:  vary the scaffold, the binder
size within the band, and the epitope.  Do not emit the same
specification twice.

Express every design only through channels the method reference
defines, using only values from its closed sets.
```

H.2.5 CD45 SPECIFICATION EXAMPLES

All three expansion runs received the same CD45 research dossier and target structure. Claude generated each method's designs in batches and recorded a rationale for every design. The examples below pair verbatim excerpts from these saved rationales with the corresponding input sent to each structure model. The three expansion runs proposed designs independently, so these examples illustrate the available conditioning channels rather than matched design strategies.

Prompteina. For `ra_cd45_114`, Claude's batch-level analysis identifies the "N-terminal face / BC8 epitope" as a mixed charged and polar site and proposes "VHH-inspired loop-mediated contacts to [the] N-terminal face." The natural-language specification sent to Prompteina retains the proposed fold, interface size, buried area, chemistry, and geometry:

```
Structural class:  Mainly Beta
Architecture:  Sandwich
Topology:  Immunoglobulin-like
Strands:  9
Secondary structure:  45% strand, 55% coil
Protein-protein interface:  26 interface residues
Buried surface area:  1800 A^2
Interface H-bonds:  10 total
Interface hydrophobic contacts:  12 total
Interface geometry:  concave, heterologous, mixed
```

RFdiffusion3. For one design, Claude records the following rationale:

> "Engages the large polar surface patch on Face A (Q36, N37, T39, N75) with directed hydrogen bonds. This is the most polar-dense contiguous surface on d1, and a helical binder presenting complementary polar side chains should achieve specificity through a network of H-bonds. The acceptor on Q36 and donor on N75 anchor the interface polar core."

The emitted JSON retains a non-loop flag, the selected target atoms, and two atom-level hydrogen-bond labels:

```
is_non_loopy:  true
select_hotspots:  A36, A37, A39, A75
select_hbond_acceptor:  A36.OE1
select_hbond_donor:  A75.ND2
```

Across the 200 CD45 specifications, 193 set the non-loop flag, 38 specify a hydrogen-bond acceptor, 17 specify a hydrogen-bond donor, and 7 contain an atom-level burial or exposure label.

BoltzGen. For another design, Claude records the following rationale:

> "Targets the MFI hydrophobic patch (M47, I48, F49) in loop 3, a compact surface surrounded by polar/charged residues (R41, Q43, N46, D50, N51, K52). A mixed α/β topology uses a β-hairpin to extend along the loop 3 edge for shape complementarity, while two helices provide hydrophobic packing against M47/I48/F49 and additional polar contacts."

The emitted YAML retains residue-level secondary-structure ranges and target binding residues:

```
helix:  27..48, 55..76
sheet:  1..8, 13..20
loop:  9..12, 21..26, 49..54, 77..82
binding:  41, 43, 46, 47, 48, 49, 50, 51, 52
```

All 200 BoltzGen CD45 specifications contain residue-level secondary-structure ranges. These examples show that Claude considered site chemistry and geometry for all three methods, while each model's input interface determined which parts of that reasoning reached generation.

H.3 Baseline agent specifications

The released Complexa checkpoint takes the target structure, binder length, and target hotspots. Its authors also report a CATH-conditioned binder checkpoint, but have not released it. RFdiffusion3 specifications can set binder length and chain layout, atom-level target hotspots, hydrogen-bond partners, burial state, and non-loop secondary structure. BoltzGen specifications can set binding and avoidance sites, per-residue secondary-structure ranges, and permitted amino acids.

Two protocol differences determine how the agentic and non-agentic runs should be compared. First, the binding site comes from a different source. Non-agentic runs receive expert-curated hotspots from the AlphaProteo (Zambaldi et al., 2024) and BindCraft (Pacesa et al., 2025) campaigns on the 18 targets that provide them. Agentic runs instead select an epitope for every design, including Prompteina's designs.

Second, the agent populates the baselines' conditioning channels instead of leaving them at their defaults. A complete parse of all 8,200 specifications per method quantifies this use (Table 20). The eight RFdiffusion3 specifications without an epitope use center-of-mass strategies, for which the agent specifies an inferred orientation instead of a hotspot.

Table 20: **Baseline conditioning-channel use.** Coverage is the fraction of that method's 8,200 specification files that populate each channel.

Method	**Channel**	Coverage
RFdiffusion3	Target hotspots (`select_hotspots`)	99.9%
	Non-loop secondary structure (`is_non_loopy`)	93.8%
	Hydrogen-bond acceptors	18.5%
	Hydrogen-bond donors	8.0%
	Burial, buried (`select_buried`)	2.0%
	Burial, partial (`select_partially_buried`)	2.0%
	Burial, exposed (`select_exposed`)	0.8%
BoltzGen	Secondary structure, per residue	100.0%
	Loop assignments	99.3%
	Negative design (`not_binding`)	84.9%
	Sheet assignments	14.7%
	Permitted amino acids	0.4%

H.4 Binder score calibration

AlphaProteo AF2 filters used by Complexa. AlphaProteo fit three AlphaFold2 thresholds to the binding data of Cao et al. (2022): interaction PAE below 7 Å, complex pLDDT above 0.90, and binder-aligned RMSD below 1.5 Å (Zambaldi et al., 2024). The thresholds were selected by ranking experimental success rates within each target, subject to at least ten designs passing the filters. Complexa subsequently adopted these AF2 filters as its in-silico success criterion (Didi et al., 2026). AlphaProteo instead used its AF3 filters for its final benchmark because they performed

better across its targets. These differences motivate recalibrating the evaluation criterion for method comparison.

Calibration data. We assemble 5,178 designed binders with experimental bind/no-bind labels from four public releases: 3,532 designs over 13 targets from Overath et al. (2025), 1,140 over 15 targets from Shanehsazzadeh (2026), 361 over 13 targets from the BoltzGen deposit (Stark et al., 2026), and 145 over 13 targets from the BindCraft deposit (Pacesa et al., 2025). We treat each release–target pair as a fold unit because releases with the same target name can use different constructs and design populations. Of 54 fold units, 46 contain both outcome classes and can be used to fit a threshold. We therefore estimate uncertainty over 46 fold units rather than 5,178 designs.

Prediction and calibration protocol. We refold every design with one seed under ESMFold2-Fast, ESMFold2, AlphaFold2, and AlphaFold3 in a single evaluation harness. AlphaFold2 uses initial guess, the standard binder-scoring configuration that initializes the structure module from the designed coordinates. This common harness avoids confounding prediction-model differences with laboratory-specific configurations in the published score columns. We recompute every AlphaProteo AF2 filter from the persisted structures and predicted aligned error matrices under all four models. Thresholded criteria use leave-one-fold-unit-out evaluation, whereas average precision is threshold-free and requires no cross-validation.

Table 21: **Binder-score calibration.** Metrics are macro-averaged over fold units. Frozen criteria use their published thresholds, and tuned criteria use leave-one-fold-unit-out evaluation. The macro chance level is 0.147.

	ESMFold2-Fast	**ESMFold2**	**AF2**	**AF3**
Macro-F1				
AlphaProteo AF2 filters	0.255	0.226	0.297	0.389
its term iPAE < 7 alone	0.452	0.336	0.310	0.389
its term pLDDT > 0.90 alone	0.310	0.188	0.262	0.402
its term scRMSD < 1.5 alone	0.241	0.159	0.295	0.279
iPAE, tuned threshold	0.409	0.349	0.322	0.384
$\text{ipSAE}_{\min}$, tuned threshold	**0.551**	0.446	0.356	0.493
Macro average precision				
iPAE	0.559	0.483	0.362	0.563
$\text{ipSAE}_{\min}$	**0.594**	0.519	0.386	**0.596**
Seconds per design	**1.61**	2.37	2.62	9.07
Target coverage at $\text{ipSAE}_{\min}$	**13/13**	12/13	13/13	13/13

AlphaProteo AF2 filter evaluation. Under every prediction model, the AlphaProteo AF2 filter set discriminates binders less accurately than any one of its component thresholds (Table 21). The conjunction does not produce a positive prediction for every target. Under AlphaFold2, the published form returns a positive prediction on only 12 of 13 targets, and the commonly used substitution of binder pLDDT and target-aligned RMSD reduces coverage to 9. A criterion that returns no positives for a target cannot compare methods on that target. These comparisons use the published thresholds and do not depend on retuning.

$\text{ipSAE}_{\min}$ evaluation. $\text{ipSAE}_{\min}$ restricts a TM-score-like transform of predicted aligned error to residue pairs across the interface and scales its distance parameter by interface size (Shanehsazzadeh, 2026). The *min* variant takes the smaller of the asymmetric binder→target and target→binder scores. Its macro average precision exceeds that of iPAE under every prediction model (Table 21). Adding an RMSD threshold does not improve it; for example, pairing $\text{ipSAE}_{\min}$ with self-consistency RMSD reduces macro-F1 (0.551 to 0.267).

ESMFold2-Fast and AlphaFold3 give macro average precision values of 0.594 and 0.596, with runtimes of 1.61 and 9.07 seconds per design, respectively; the benchmark therefore uses ESMFold2-Fast. We report continuous $\text{ipSAE}_{\min}$ directly because calibrated thresholds inherit the population on which they were fitted. Within each target, we average the k highest-scoring designs to measure the part of the distribution that a design campaign would order. We then average these target sum-

maries, retaining the target as the sampling unit. Every method contributes 200 designs on each of the same 20 targets.

Let $M(k)$ be the mean $\text{ipSAE}_{\min}$ of a target's k best designs among the N generated designs. The expected top-k statistic averages $M(k)$ under a log-uniform distribution on k,

$$\mathbb{E}\big[M(K)\big] = \left(\sum_{k=1}^{N} \tfrac{1}{k}\right)^{-1} \sum_{k=1}^{N} M(k)/k. \tag{10}$$

This weighting gives equal mass to each order of magnitude of the design budget.

Limitations. Published binder thresholds, including ours, are fitted on partially overlapping data; 2,585 of the 3,532 designs in our largest release come from Cao et al. (2022). The Shanehsazzadeh (2026) release was selected using $\text{ipSAE}_{\min}$, so any threshold fitted on that release inherits its selection. We therefore emphasize the threshold-free ranking statistic. All calibration targets are single chains, whereas three benchmark targets are not. We apply one chain-merge convention consistently, but do not measure its effect on transfer. The negative examples come from real design campaigns and are therefore near-misses rather than random sequences. This population is appropriate for discriminating plausible designs but cannot determine an absolute experimental hit rate.

H.5 Prediction-model replication

Table 22: **AlphaFold3 replication.** Expected top-k $\text{ipSAE}_{\min}$ is computed from the same 200 designs per method–setting combination on each of the 20 targets. Only the `self` condition was refolded; `MPNN` and `MPNN-FI` were not evaluated with AlphaFold3. AlphaFold3 uses one seed and `num_recycles=10`, versus three recycles in Table 21.

Method	**Agent**	ESMFold2-Fast	AlphaFold3
Prompteina	✓	**0.455**	**0.518**
Complexa	✗	<u>0.400</u>	<u>0.434</u>
BoltzGen	✓	0.359	0.420
BoltzGen	✗	0.376	0.404
RFdiffusion3	✓	0.237	0.276
RFdiffusion3	✗	0.251	0.293

Prompteina remains first under AlphaFold3, with a larger margin over Complexa than under ESMFold2-Fast (0.084 vs. 0.055). The ordering of agent-guided and non-agentic BoltzGen reverses between folding models, so we do not interpret the direction of that comparison.

H.6 Predicted binding across design budgets

We compare predicted binding across the full range of retained design budgets in Table 23. Under `self` and `MPNN-FI` Prompteina leads at every budget. Under `MPNN`, Prompteina leads only in the middle of the range; RFdiffusion3 leads for $k \leq 2$ and BoltzGen leads for $k \geq 64$. No method therefore leads at every budget in this sequence condition. The k=1 column rests on one design per target and separates the method–setting combinations least reliably.

H.7 Generated binder–target interface properties

We analyze generated-backbone properties to isolate the design model's contribution, then measure the same properties after refolding to test whether they persist under sequence redesign. The three refolds use the generated sequence, the best full ProteinMPNN redesign, or the best fixed-interface redesign. Free redesign raises the baselines' median interface hydrogen-bond counts by one to three, whereas self-sequence refolding and fixed-interface redesign leave them near their generated values (Table 24). Buried surface area follows the same pattern, with Prompteina's median advantage over the best baseline narrowing as ProteinMPNN receives more freedom (149 Å^2 on generated backbones, 99 Å^2 under `MPNN-FI`, and 55 Å^2 under full `MPNN` redesign).

Holding the agent fixed, Prompteina generates larger binder–target interfaces than the structured-conditioning baselines (median buried surface area 1,020 Å^2 vs. 710–871 Å^2; Table 25). Its

Table 23: **Predicted binding across design budgets.** Each cell is the within-target mean of the target's k best ipSAE$_{\min}$ values, averaged over 20 targets; the last column is the expected top-k from Eq. 10. Every method–setting combination contributes 200 designs per target, so k=200 is the mean over all designs. Scores use ESMFold2-Fast at one seed. **Bold** marks values tied for the column maximum at displayed precision; at `MPNN` k=4, RFdiffusion3 exceeds Prompteina by 0.0003 although both round to 0.689.

Method	Agent	k=1	2	4	8	16	32	64	128	200	$\mathbb{E}[M(K)]$
`self` — each model's own sequence											
Prompteina	✓	**0.647**	**0.620**	**0.580**	**0.534**	**0.470**	**0.376**	**0.253**	**0.149**	**0.101**	**0.455**
Complexa	✗	0.571	0.543	0.501	0.455	0.402	0.333	0.240	0.144	0.096	0.400
BoltzGen	✓	0.573	0.537	0.478	0.405	0.326	0.249	0.169	0.107	0.074	0.359
BoltzGen	✗	0.569	0.530	0.486	0.430	0.364	0.284	0.199	0.128	0.088	0.376
RFdiffusion3	✓	0.475	0.425	0.355	0.258	0.161	0.093	0.051	0.026	0.017	0.237
RFdiffusion3	✗	0.461	0.426	0.362	0.286	0.200	0.125	0.071	0.036	0.023	0.251
`MPNN` — free ProteinMPNN redesign, 8 sequences per design											
Prompteina	✓	0.720	0.709	**0.689**	**0.660**	**0.613**	**0.545**	0.445	0.308	0.212	**0.584**
Complexa	✗	0.680	0.661	0.634	0.600	0.553	0.489	0.405	0.290	0.200	0.538
BoltzGen	✓	0.729	0.714	0.684	0.646	0.592	0.522	0.421	0.296	0.204	0.575
BoltzGen	✗	0.711	0.695	0.668	0.636	0.594	0.535	**0.454**	**0.335**	**0.234**	0.577
RFdiffusion3	✓	**0.759**	**0.726**	**0.689**	0.641	0.578	0.489	0.363	0.220	0.143	0.559
RFdiffusion3	✗	0.698	0.680	0.658	0.624	0.581	0.521	0.434	0.320	0.218	0.563
`MPNN-FI` — redesign with the generated interface fixed											
Prompteina	✓	**0.696**	**0.679**	**0.659**	**0.629**	**0.582**	**0.503**	**0.394**	**0.256**	**0.174**	**0.549**
Complexa	✗	0.658	0.638	0.612	0.576	0.528	0.462	0.367	0.239	0.163	0.510
BoltzGen	✓	0.674	0.652	0.620	0.569	0.498	0.406	0.299	0.194	0.133	0.490
BoltzGen	✗	0.659	0.631	0.597	0.561	0.508	0.434	0.332	0.220	0.152	0.493
RFdiffusion3	✓	0.675	0.642	0.599	0.537	0.459	0.349	0.229	0.127	0.082	0.454
RFdiffusion3	✗	0.664	0.635	0.602	0.548	0.475	0.383	0.272	0.164	0.106	0.469

Table 24: **Interface hydrogen bonds after sequence redesign.** Values are median interface hydrogen-bond counts per design, measured on the generated backbone and on the structure folded under each sequence condition. Agentic and non-agentic runs are separate.

Method	Agent	Generated	`self`	`MPNN`	`MPNN-FI`
Prompteina	✓	**10**	**8**	**9**	**9**
Complexa	✗	7	7	8	8
BoltzGen	✓	6	6	7	6
BoltzGen	✗	7	7	8	7
RFdiffusion3	✓	5	6	8	6
RFdiffusion3	✗	6	5	**9**	6

ipSAE$_{\min}$ rises across within-target buried-surface-area quartiles at every budget (Q1 to Q4: 0.280 to 0.676 at k=1, 0.016 to 0.421 at k=32, and 0.070 to 0.207 over all 200 designs). Among generated-backbone properties, buried surface area has the strongest mean within-target correlation with the `MPNN-FI` score (0.18 across the six method–setting combinations, compared with 0.14 for interface residue count and 0.09 for interface hydrogen bonds). The association holds for every method except RFdiffusion3 when the generated interface is preserved (Spearman ρ 0.26 for Complexa, 0.22–0.23 for BoltzGen, 0.18 for Prompteina, and 0.07–0.09 for RFdiffusion3). Under free `MPNN` redesign, RFdiffusion3's correlation rises to 0.24–0.27 as ProteinMPNN rebuilds interface contacts.

Matching designs on generated buried surface area changes Prompteina's lead only slightly (13 to 12 of 20 targets over BoltzGen and 17 to 18 of 20 over RFdiffusion3). Matching on generated interface hydrogen bonds also retains the RFdiffusion3 gap (16 of 20 targets), as do the best-design score and the number of designs above 0.5 (mean best score 0.696 vs. 0.675; median yield 22 vs. 8 per target). Buried surface area therefore explains only part of the difference; on unseen targets, no combination of measured per-design properties predicts ipSAE$_{\min}$ beyond $r \approx 0.4$.

Table 25: **Generated interface properties and predicted binding.** All three methods use the same agent, and properties are measured on the generated backbone. Buried-surface-area quartiles are defined within each target across all three methods. The final row reports the `MPNN-FI` score in the lowest and highest quartiles.

	Prompteina (text)	**BoltzGen** (YAML)	**RFdiffusion3** (JSON)
Realized properties			
Buried surface area (Å^2, p10/med/p90)	**691/1020/1422**	468/728/1073	400/710/1205
Designs in the top quartile of area	**47%**	11%	17%
ipSAE$_{\min}$ *across quartiles of buried surface area*			
Q1 (508 Å^2) → Q4 (1175 Å^2)	0.116 → **0.596**	0.279 → 0.442	0.361 → 0.313

H.8 Binder specification ablation

We re-run the agentic pipeline of §3.2.2 with a reduced specification interface to measure the contribution of agent-specified properties. At each level the agent plans from scratch: it reads the same research dossier, but the design properties it may write are restricted, and it chooses its own epitope and binder length under that restriction. Fold-only exposes structural class, architecture and topology. Fold + shape adds secondary structure, helix and strand counts, compactness, surface geometry, contact order, charge character and surface hydrophobicity. The default interface adds interface residue count, buried surface area, interface geometry, hydrogen bonds, hydrophobic contacts and salt bridges.

Each level generates 200 designs per target on the same 20 targets, and we score the generated sequence, since the ablation targets what conditioning produces rather than what ProteinMPNN recovers. The default level is the Prompteina row of Table 3 and is not regenerated.

Removing the interface properties costs 0.091 expected top-k (95% CI $[0.041, 0.145]$, paired bootstrap over the 20 targets), and the full specification leads at every budget (Table 26). Removing monomer shape on top of that costs a further 0.002 ($[-0.036, 0.037]$), so the two reduced levels are not separable from each other; the effect is a single step at the interface properties rather than a graded response to specification length. Across targets, 9 of 20 realize the full ordering against a chance level of 3.3 under a random ordering of three levels (Binomial$(20, 1/6)$, $p = 0.003$), and on all 9 the end-to-end difference is resolvable within the target. Two targets reverse it resolvably (CrSAS6 and Insulin) and five are not separable.

A drop could in principle reflect a worse epitope choice rather than the loss of the properties themselves because each level re-plans. We therefore also ran a variant that holds the epitope and binder length fixed at the values chosen under the full interface and removes only the text, which gives 0.094 for the same contrast. That variant is a lower bound, since a plan authored under the full property set still encodes the removed intent in its epitope; its agreement with the re-planning result to within 0.004 indicates that epitope choice does not account for the effect.

Table 26: **Binder specification ablation.** Each cell is the within-target mean of the target's k best ipSAE$_{\min}$ values under the generated sequence, averaged over 20 targets; the last column is the expected top-k from Eq. 10. Every level contributes 200 designs per target, and the agent re-plans at each level. Parentheses list the design properties the agent may specify. Scores use ESMFold2-Fast at one seed. The two reduced levels are not separable from each other at any budget.

Specification interface	k=8	16	32	64	128	$\mathbb{E}[M(K)]$
Prompteina (default: fold + shape + interface)	**0.534**	**0.470**	**0.376**	**0.253**	**0.149**	**0.455**
Prompteina (fold + shape)	0.430	0.356	0.267	0.176	0.105	0.364
Prompteina (fold only)	0.412	0.336	0.250	0.164	0.101	0.362

H.9 Structural diversity

Diversity is measured on the same design pools as binding quality, allowing the two quantities to be compared at a common selection budget. For each method and target, we rank 200 designs by ipSAE$_{\min}$, retain the best k, remove the target chain, and align every binder pair with exact TMa-

lign. Pair similarity is $\mathrm{TM} = \min(\mathrm{qTM}, \mathrm{tTM})$, where lower similarity indicates greater structural diversity. TM-score depends on length, so we compare binders only within the same 25-residue length bin and average first within bins, then across bins, and finally across targets.

Designs with $\mathrm{ipSAE}_{\min} = 0$ have an interface but no confident inter-chain geometry, and deep top-k pools can otherwise be dominated by ties at zero. We report a target at a given k only when every method has at least k positive-scoring designs. At $k = 32$, full and fixed-interface redesign retain 19 and 18 of 20 targets, respectively. We emphasize $k = 32$ because $k = 64$ retains only 17 and 13 targets, respectively, and preferentially drops difficult ones, changing panel difficulty as well as size.

Table 27: **Structural diversity among high-scoring designs.** Values are mean pairwise TM-scores among each method's k best designs under full and fixed-interface ProteinMPNN redesign, where lower is more diverse. The four columns retain 20, 19, 19, and 17 targets under full redesign and 19, 19, 18, and 13 under fixed-interface redesign.

Method	**Agent**	k=8	16	32	64
Full ProteinMPNN redesign					
Prompteina	✓	0.433	0.441	0.424	0.423
Complexa	✗	0.492	0.492	0.467	0.478
BoltzGen	✓	0.480	0.486	0.496	0.496
BoltzGen	✗	0.635	0.653	0.619	0.612
RFdiffusion3	✓	0.418	0.412	0.414	0.414
RFdiffusion3	✗	0.482	0.470	0.461	0.466
Fixed-interface ProteinMPNN redesign					
Prompteina	✓	0.455	0.438	0.416	0.410
Complexa	✗	0.536	0.504	0.480	0.468
BoltzGen	✓	0.463	0.476	0.484	0.494
BoltzGen	✗	0.640	0.616	0.601	0.630
RFdiffusion3	✓	0.434	0.413	0.413	0.412
RFdiffusion3	✗	0.459	0.454	0.459	0.451

The pooled mean and per-target comparison differ for agent-guided RFdiffusion3, which has the lowest pooled similarity but splits the targets almost evenly with Prompteina. We therefore report both summaries. At $k = 32$, Prompteina is more diverse under full and fixed-interface redesign on 13 of 19 and 15 of 18 targets against Complexa, 16 of 19 and 15 of 18 against agent-guided BoltzGen, 19 of 19 and 18 of 18 against non-agentic BoltzGen, and 13 of 19 and 14 of 18 against non-agentic RFdiffusion3, respectively. It splits the targets with agent-guided RFdiffusion3 under both conditions (8 of 19 and 9 of 18). Prompteina's binding-quality advantage therefore does not correspond to lower structural diversity in these top-ranked pools.

Table 28: **Secondary-structure composition among high-scoring designs.** Statistics use the same top-32 `MPNN-FI` pools, computed within each target and averaged over the 18 retained targets. No column has a preferred direction. Responsiveness is the standard deviation of a method's per-target mean helix fraction, so a low value indicates little change among targets.

		Helix fraction					
Method	**Agent**	p10	p50	p90	Mainly helical	Strand segments	Respon-siveness
Prompteina	✓	0.312	0.729	0.874	45%	0.7	0.127
Complexa	✗	0.278	0.565	0.777	27%	2.3	0.201
BoltzGen	✓	0.664	0.823	0.880	68%	0.0	0.035
BoltzGen	✗	0.163	0.406	0.764	29%	5.0	0.227
RFdiffusion3	✓	0.320	0.753	0.866	50%	0.5	0.120
RFdiffusion3	✗	0.529	0.796	0.872	63%	0.2	0.102

Agent-guided BoltzGen varies least among targets in secondary-structure composition (responsiveness 0.035 vs. 0.201 and 0.227 for the two non-agentic methods; Table 28). It produces no designs below 0.20 helix fraction and has a median of zero strand segments. It is also the only method whose pairwise similarity increases with k (0.463 at $k = 8$ to 0.494 at $k = 64$), indicating that its additional designs repeat similar folds.

Secondary-structure composition and structural diversity measure different variation. Non-agentic BoltzGen has the widest helix spread, the most strand segments, and the most β-containing designs, yet also the highest mean pairwise TM-score. A method can therefore vary secondary-structure content while emitting similar overall folds.

H.10 ITERATIVE PROMPT OPTIMIZATION

Each iterative loop uses the same 200-design budget as the one-shot benchmark, divided into four rounds of 50 (§3.2.2). The first round follows the one-shot agentic protocol, so iterative and independent generation differ only after the first 50 designs are scored (App. H.2). Generation, redesign, and scoring otherwise remain identical.

Each 50-design round divides the target's binder-length range into 10 bands and draws five designs per band. We recompute these bands from the full range instead of merging the 20-band grid used for 200 designs, reproducing the length distribution of an independent 50-design run.

Feedback. After each round the agent receives every design produced so far, ranked by the objective

$$y = \max\big(\mathrm{ipSAE}_{\min}(\texttt{self}),\ \max_j \mathrm{ipSAE}_{\min}(\texttt{MPNN}, j),\ \max_j \mathrm{ipSAE}_{\min}(\texttt{MPNN-FI}, j)\big),$$

maximized over the eight redesigned sequences j for each design. Each feedback entry contains the binder length, epitope, generating specification, per-condition scores, and the maximum and median across redesigned sequences. Reporting both distinguishes a design with one favorable sequence, such as maximum 0.53 and median 0.05, from a backbone that tolerates redesign, such as maximum 0.40 and median 0.33.

The feedback contains no summary statistics, and the agent is asked to draw its own conclusions. Roll-ups by design property are omitted because they are confounded by length: ranking architectures by mean score promotes those that happen to occur in long binders, and architecture does not predict the score once length is accounted for.

Rounds. A single agent call plans all 50 designs in each round. Parallel calls would receive the same prompt without seeing one another's choices and can therefore produce overlapping designs. Planning the round jointly also lets the agent distribute designs across binding sites and scaffolds.

Claudin-1 trajectory. The seed, round-1, and round-2 designs compared in Fig. 5 use the same six-residue epitope, up-down bundle architecture, convex mixed interface, and two requested intra-chain disulfides. The complete reasoning and planning passages excerpted in the figure are reproduced below.

> *Seed to round 1—reasoning.* "Pulling together the key patterns across all designs: Up-down Bundle is clearly the dominant successful architecture, the helical face epitope scored best overall at 0.667, small ECL1 binders work well for site B, larger Up-down Bundles suit the strand-helix junction, disulfide bonds seem to boost binding, and convex/mixed interface geometries outperform flatter or more hydrophobic ones, with moderate interface sizes around 20–22 residues looking optimal."
>
> *Seed to round 1—planning.* "I'm sketching 15 designs for this group, staying close to the winning design while probing topology and parameter variations: several PLA2-based scaffolds at different sizes (125–140 residues) with convex or flat mixed/hydrophobic surfaces and 2 disulfides, plus alternative topologies using colicin Ia, a smaller 4-helix bundle, acyl-CoA binding protein, growth hormone, and bacteriocin As-48 scaffolds, each tested with similar disulfide and interface parameters."
>
> *Round 1 to round 2—reasoning.* "Comparing #2 vs #4, the convex geometry with more interface residues, BSA, and H-bonds outperforms the flat version. And comparing #1 vs #2, growth hormone's 4-helix topology edges out PLA2's 5-helix design despite fewer interface residues and less BSA, suggesting topology choice matters as much as raw interface size."
>
> *Round 1 to round 2—planning.* "I should focus most designs on refining around the two best epitopes: A89,A92,A93,A96,A118,A121 paired with the Growth Hormone topology (which produced the best result) and also tested with PLA2 at optimal size, plus the

> A86,A89,A92,A93,A96,A121 variant with both topologies. I'll also try small perturbations like swapping A85 or A86 in, and tune interface parameters toward convex geometry with 20–24 interface residues around 1500–1700."

The selected round-1 design changes the requested topology from phospholipase A2 to growth hormone. The selected round-2 design keeps that topology and increases the requested interface from 20 to 24 residues and from 1500 to 1700 Å^2 buried surface area (Table 29).

Table 29: **Claudin-1 trajectory under iterative prompting.** Generated properties are measured directly on model outputs; ipSAE$_{\min}$ is computed on their ESMFold2-Fast refolds.

Property	Initial	Round 1	Round 2
Requested specification			
Topology	Phospholipase A2	Growth hormone	Growth hormone
Helices	5	4	5
Interface residues	22	20	24
Buried surface area (Å^2)	1600	1500	1700
Generated structure			
Binder length	135	136	141
Helices (DSSP)	7	4	4
Interface residues	32	21	33
Buried surface area (Å^2)	1298	898	1327
ipSAE$_{\min}$	0.667	0.737	0.802

Claudin-1 comparison. Figure 5a shows the top-1 gain; the same comparison at top-8 gives the same method ordering (Table 30).

Table 30: **Iterative improvement on Claudin-1.** Each value is the gain in cumulative top-k mean ipSAE$_{\min}$ over the exact expectation from a matched-budget i.i.d. extension of the shared 50-design seed.

		Designs evaluated			
Method	k	50	100	150	200
Prompteina	1	0.000	+0.060	+0.114	+0.119
	8	0.000	+0.121	+0.167	+0.181
BoltzGen	1	0.000	−0.002	+0.019	+0.029
	8	0.000	+0.013	+0.051	+0.056
RFdiffusion3	1	0.000	+0.002	−0.012	+0.013
	8	0.000	+0.048	+0.035	+0.041

Baseline reasoning traces. When paired with BoltzGen or RFdiffusion3, the agent also extracts coherent patterns from feedback; both iterative runs improve over matched-budget i.i.d. generation on Claudin-1, although by less than Prompteina. Before round 2, the BoltzGen agent concludes that "Helix-βhairpin-helix (H-SS-H-H) dominates" and assigns 70% of the next batch to variants of that topology at the identified length range. Before round 3, the RFdiffusion3 agent finds that "The A47+A67 core (ECL1 β-sheet hydrophobic crown) dominates all top scores" and assigns about 70% of the next batch to systematic hotspot and length sweeps around that site. Thus, the smaller baseline gains do not reflect a failure to use the feedback. These single-target traces suggest that Prompteina gives the agent more ways to act on feedback: it can revise topology, architecture, interface geometry and size, and disulfides, whereas through BoltzGen and RFdiffusion3 it mainly revises secondary-structure ranges, atom-level target constraints, and binder length.

H.11 Full-panel iterative prompt optimization

We run the same iterative loop on all 20 targets to test whether these gains extend across the full benchmark. Each method receives 200 designs per target.

We define target difficulty as the mean expected top-k ipSAE$_{\min}$ of the three i.i.d. runs on that target, averaged over the three sequence-redesign conditions. A lower value indicates that every method performs worse on the target. Difficulty is a property of the i.i.d. runs alone, so the loop cannot move it. IL-17A and TNF-α rank second and third hardest of the 20. We split the panel into difficulty thirds and report each method's gain over its own i.i.d. pool under the same top-k metric (Table 31).

Independent difficulty check. An independent check separates the difficulty definition from the gain estimate. We define difficulty using a deterministic, length-stratified subset of 50 i.i.d. designs per method and target, then evaluate gains using the disjoint remaining 150 i.i.d. designs and 150 agent-generated designs from rounds 1–3. This independent definition produces exactly the same hardest, middle, and easiest target sets. We use this check only to verify the target grouping rather than to compare gain magnitudes because it changes the evaluation budget and removes the shared seed used in the primary analysis.

Prompteina gains most on the hardest third at both k=1 and k=8, and its gains there exceed those of the other methods. BoltzGen follows the same pattern at k=1 but gains most on the middle third at k=8. RFdiffusion3 loses ground on the hardest targets at k=1 and is at chance across the panel; its gain comes from the easier two thirds and from k=8 rather than its best design.

Table 31: **Full-panel iterative prompting by target difficulty.** Every method receives 200 designs per target across the 20-target panel. Each cell is the mean gain in top-k ipSAE$_{\min}$ over the same method's 200 i.i.d. designs, averaged over the targets in that difficulty third. Targets improved counts those for which the loop beats i.i.d. generation. **Bold** marks the best method within a column.

		Gain over i.i.d. (↑)			Targets
Method	k	Hardest 7	Middle 6	Easiest 7	improved
Prompteina	1	+**0.098**	+**0.046**	+**0.014**	**17/20**
BoltzGen	1	+0.051	+0.042	+0.012	16/20
RFdiffusion3	1	−0.021	+0.004	+0.003	11/20
Prompteina	8	+**0.105**	+0.073	+0.023	**20/20**
BoltzGen	8	+0.067	+**0.087**	+0.022	**20/20**
RFdiffusion3	8	+0.028	+0.055	+**0.030**	16/20

The hardest third contains CbAgo, IL-17A, TNF-α, BBF14, CD45, HER2, and SpCas9. The middle third contains Claudin1, DerF7, CrSAS6, VEGFA, DerF21, and BetV1. The easiest third contains TrkA, IL7RA, IFNAR2, PD1, Insulin, PD-L1, and BHRF1. Ranking the targets by top-1 or top-8 rather than expected top-k leaves the six hardest unchanged. At most six targets cross a boundary, all at the middle-to-easiest edge where the difficulty values are closely spaced.

H.12 PROSPECTIVE GM2A BINDING ASSAY

We ran four parallel iterative-optimization threads (§3.2.3), each targeting a distinct GM2A epitope, with one additionally constrained to β-rich binder topologies. After ranking candidates using ESMFold2-Fast and AlphaFold3, we selected 90 designs for experimental evaluation by Adaptyv Bio. Of the 28 designs loaded onto the SPR assay, one was classified as a binder based on a concentration-dependent response with discernible association and dissociation phases.

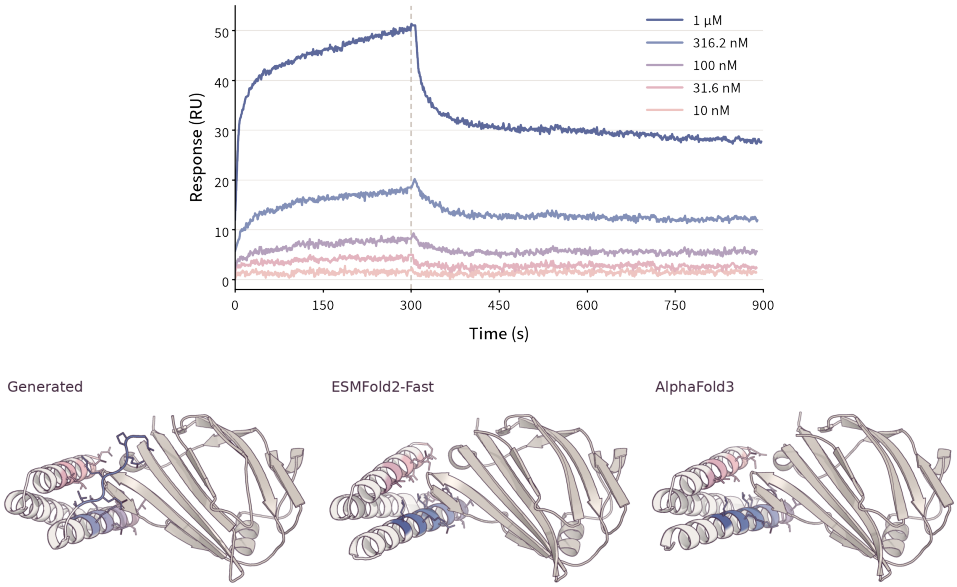


Figure 8: **Experimental signal and structural models for the candidate GM2A binder. (a)** Measured response curves at five GM2A concentrations; the dashed line at 300 s marks the transition from association to dissociation. At 1 μM, the response reaches approximately 50 RU at this boundary and remains near 28 RU after 600 s. Data are from one replicate and shown without a kinetic fit. **(b)** Generated, ESMFold2-Fast, and AlphaFold3 structures of the same design, aligned on GM2A. GM2A is shown in gray, with residues at the predicted binder–target interface highlighted.