RosettaSearch: Multi-Objective Inference-Time Search for Protein Sequence Design with LLMs

Meghana Kshirsagar [1] **Allen Nie** [2] **Ching-An Cheng** [3] **Fanglei Xue** [4] **Rahul Dodhia** [5] **Juan Lavista Ferres** [5] **Kevin K. Yang** [6] **Frank DiMaio** [4]

Abstract

We introduce RosettaSearch, an inference-time multi-objective optimization approach that instantiates a closed-loop framework for the refinement of backbone-conditioned protein sequences. At each iteration, structured rewards and residue-level feedback from RosettaFold3 drive targeted sequence refinement across multiple candidate trajectories in parallel, within a strictly bounded budget of 75 structure prediction calls per design task. In a large-scale evaluation on ≈400 protein redesign tasks, RosettaSearch achieves 18–68% improvements in structural fidelity metrics over sequences generated by LigandMPNN, translating to a $2.5\times$ improvement in design success rate. Gains are robust under an independent structure predictor (Chai-1) and generalize across two LLM families (o4-mini and Gemini-3), with performance scaling consistently with reasoning capability. We further extend the framework to vision-language models, where rendered images of predicted structures provide spatial feedback that produces richer structural reasoning in the model's chain-of-thought. To our knowledge, this is the first large-scale demonstration that LLMs can serve as effective generative optimizers for backbone-conditioned protein sequence design, yielding systematic gains without any model retraining or fine-tuning.

[1]Calico Life Sciences (prior work before joining) [2]Google DeepMind (prior work before joining) [3]Google Research (partial work done previously at Microsoft Research) [4]Institute for Protein Design, University of Washington [5]AI for Good, Microsoft, Redmond [6]Microsoft Research, New England. Correspondence to: Meghana Kshirsagar <meghana.ksagar@gmail.com>.

Accepted at ICML 2026 AI for Science Workshop, Seoul, South Korea. Copyright 2026 by the author(s).

1. Introduction

Protein sequence design—the task of identifying amino acid sequences that fold into a target three-dimensional structure with desired functional properties—is a foundational challenge in computational biology with broad implications for drug discovery, enzyme engineering, and the design of novel therapeutics. We focus specifically on *backbone-conditioned sequence design*: given a fixed target backbone structure, find a sequence that confidently and accurately folds into it. We use the term *fidelity* to refer to this property—whether a designed sequence, when its structure is predicted independently, recapitulates the target backbone with high confidence and geometric accuracy. Despite remarkable progress in deep learning approaches such as LigandMPNN (Dauparas et al., 2025) and ProteinMPNN (Dauparas et al., 2022), a persistent and practically significant gap remains: sequences generated by state-of-the-art models frequently exhibit low fidelity (Anishchenko et al., 2021). This matters because fidelity is the key bottleneck between computational protein design and experimental success: a sequence that cannot be reliably predicted to adopt the target structure is unlikely to do so in the wet lab, wasting costly synthesis and experimental validation effort.

The core difficulty is that backbone-conditioned sequence design is a high-dimensional, combinatorial optimization problem. The sequence space over amino acid alphabets is astronomically large, structural evaluation is expensive (requiring a full structure prediction call per candidate), and the relationship between sequence and fold is highly nonlinear, with many local optima.

Current approaches fall short in complementary ways. Single-pass sequence design models operate through a single forward pass of autoregressive decoding with no mechanism for self-correction: when a generated sequence lands in a low-fidelity region of sequence space, the model cannot recover since it cannot revisit or revise its output in response to downstream structural feedback. Optimization-based alternatives—reinforcement learning, directed evolution, and preference fine-tuning—enable iterative improvement but require task-specific training that is expensive and inflexible to new objectives.

LLMs have been explored for protein design but only in one-shot settings, producing sequences with substantially lower structural fidelity than task-specific models (Xia et al., 2025). Generative optimization with LLMs has shown promise in other discrete optimization settings such as prompt, code, and hyperparameter optimization (Khattab et al., 2023; Agrawal et al., 2025), but its potential as an optimizer for backbone-conditioned sequence design has not been systematically explored. See Section 2 for a full discussion of related work.

RosettaSearch instantiates a closed-loop design-evaluate-refine workflow in which each iteration tightens the feedback cycle between sequence generation and structural evaluation. At every step, a structure prediction call plays the role of the experimental assay in a traditional wet-lab directed evolution campaign—providing a quantitative signal that drives the next round of proposals. Unlike autonomous lab workflows that are bottlenecked by physical synthesis and measurement throughput, RosettaSearch operates entirely *in silico* and completes this loop dozens of times within a fixed budget of 75 oracle calls per design task, making it practical to run at scale across hundreds of protein targets simultaneously. The trajectories generated by this process—sequences, structural evaluations, and the reasoning traces connecting them—constitute a rich record of the search that can be repurposed as training data for the next generation of sequence design models, closing a second loop between inference-time optimization and model improvement.

We introduce **RosettaSearch** (Figure 2), an inference-time multi-objective optimization approach that addresses these limitations by deploying LLMs as generative optimizers within a structured search algorithm guided by structural rewards. Rather than asking an LLM to design a sequence from scratch, RosettaSearch instead asks it to *refine* an existing candidate by reasoning over structured feedback from RosettaFold3 (Corley et al., 2025). This feedback combines global scalar rewards with local, residue-level annotations identifying problematic regions. The LLM uses this feedback to propose targeted sequence edits within a *priority search* algorithm that explores multiple modification trajectories in parallel, avoiding premature convergence. Because the reward function and feedback can be redefined at inference time, RosettaSearch requires no retraining or fine-tuning to adapt to new design objectives.

We also extend the framework to vision-language models (VLMs), incorporating rendered images of predicted structures as feedback to provide additional spatial context, and find that this produces richer reasoning in the model's chain-of-thought. To the best of our knowledge, this is the first large-scale demonstration that LLMs can function as effective generative optimizers for backbone-conditioned sequence design, yielding systematic gains over popular baselines without retraining or task-specific fine-tuning.

2. Related Work

Backbone-conditioned sequence design. Models such as ProteinMPNN (Dauparas et al., 2022) and LigandMPNN (Dauparas et al., 2025) achieve strong sequence recovery rates but operate through a single forward pass optimized for sequence recovery rather than structural fidelity under independent evaluation. A meaningful fraction of their designs exhibit low confidence or poor geometric agreement with the target backbone, and the models have no mechanism to detect or correct such failures at inference time. Our work treats the output of such models as a starting point for iterative, feedback-driven refinement.

Protein sequence optimization. Energy-based methods (Stracquadanio & Nicosia, 2011; Zhou et al., 2024) and reinforcement learning over mutation spaces (Lutz et al., 2023) can improve sequences but typically require task-specific training. Directed evolution approaches (Yang et al., 2019) rely on random or heuristic mutation operators without structured reasoning. Preference learning approaches such as ResiDPO (Xue et al., 2025) fine-tune sequence design models using structural preference signals from AlphaFold, achieving improvements in fidelity but coupling optimization to a fixed training procedure. In contrast, RosettaSearch performs optimization entirely at inference time, requiring no retraining when objectives or design contexts change.

LLMs for protein design. Frontier general-purpose LLMs have been evaluated on protein sequence generation in one-shot settings but produce sequences with substantially lower structural fidelity than task-specific models (Xia et al., 2025). The Virtual Lab (Swanson et al., 2024) and AI co-scientist (Gottweis et al., 2025) explore multi-agent LLM workflows for protein design, but focus on orchestrating multiple trained modules rather than systematic inference-time optimization across hundreds of design problems. Our work uses the LLM as a generative optimizer that receives quantitative structural feedback and proposes targeted sequence edits within a principled search algorithm.

Generative optimization and inference-time search. A growing body of work explores LLMs as optimizers over discrete structured spaces (Khattab et al., 2023; Cheng et al., 2024; Yuksekgonul et al., 2024; Nie et al., 2024), incorporating beam search (Sun et al., 2023; Pryzant et al., 2023), MCTS (Wang et al., 2023), FunSearch (Romera-Paredes et al., 2024), and GEPA (Agrawal et al., 2025). Our work extends this line to backbone-conditioned protein sequence design, where the search space is combinatorial over a struc-

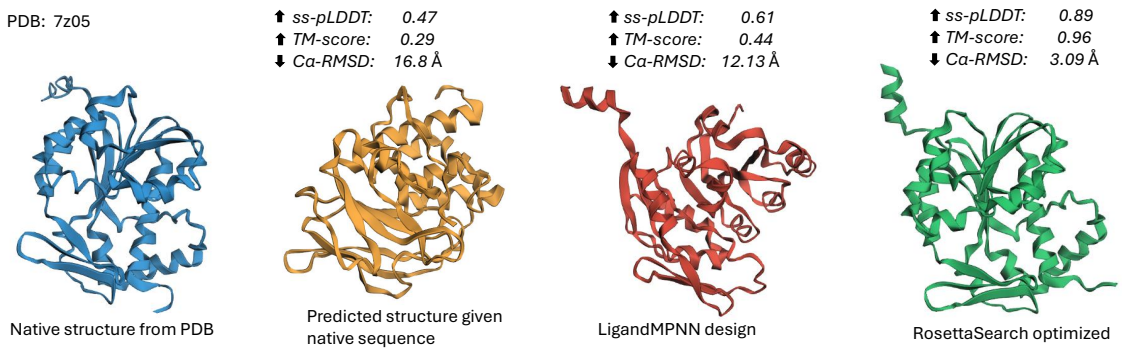


Figure 1. **Inference-time optimization with RosettaSearch dramatically improves the structural fidelity of designed protein sequences.** For a protein in our dataset (PDB: `7z05`), RosettaSearch transforms a low-fidelity LigandMPNN design into a near-native structure, achieving large gains in pLDDT, TM-score, and RMSD using only inference-time feedback, without model retraining. We show the single sequence pLDDT obtained on each predicted structure (no multiple sequence alignment, i.e. MSA input). Also shown is the TM-score obtained by comparing the predicted structure to the PDB structure in blue. *(1. Blue)* Native 3D structure downloaded from PDB. *(2. Golden)* Predicted structure given the native sequence. *(3. Red)* Predicted structure of the best design generated by LigandMPNN. *(4. Green)* Predicted structure of the best design generated by RosettaSearch. Arrows next to the metrics indicate the desired direction of improvement. See Table 10 for the amino-acid sequences corresponding to each structure. Additional examples of structures that were optimized by RosettaSearch are shown in Figure 11 (native monomers) and Figure 12 (Dayhoff atlas backbones).

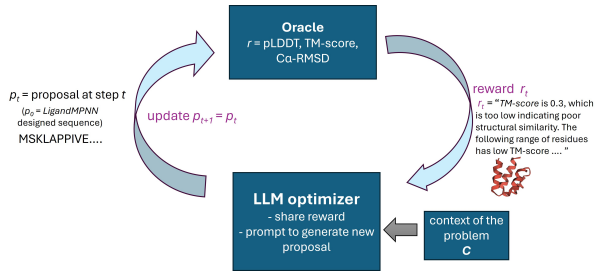


Figure 2. Schematic of the RosettaSearch approach.

tured biological alphabet, evaluation is expensive, and optimization must balance multiple structural objectives simultaneously. We are inspired by the Trace framework (Cheng et al., 2024), which we extend with a priority-based search algorithm, multi-objective reward formulation, and residue-level textual feedback.

3. Design of RosettaSearch

RosettaSearch adopts generative optimization (Khattab et al., 2023; Cheng et al., 2024) to improve protein sequence design, employing LLMs and VLMs as discrete optimization agents to analyze structural feedback and propose targeted sequence updates. This formulation optimizes the sequence via local, guided search steps rather than single-pass generation.

Notation. We use $s \in A^L$ to denote the proposed amino-acid sequence at step t, where A is the amino-acid alphabet and L is the sequence length. We use $\hat{\mathbf{x}} = f(s)$ to denote the predicted protein structure with 3D atomic coordinates, where f is the structure prediction model (RosettaFold3). We use $s^* \in A^L$ to denote a reference sequence if available, and $\mathbf{x}^*$ to denote the reference backbone's 3D atomic coordinates.

3.1. Structural fidelity metrics

Protein sequences fold to form structures, and structure determines function. We optimize three complementary fidelity metrics:

pLDDT: Per-residue confidence metric output by structure predictors such as RosettaFold3 (Corley et al., 2025) and AlphaFold3 (Abramson et al., 2024), with values $\in [0, 100]$. Higher values (>70) indicate greater confidence in local structural geometry. pLDDT correlates with local structural accuracy and serves as a proxy for fold reliability in the wet lab (Yin et al., 2022).

Cα-RMSD: Root mean square deviation between corresponding backbone α-carbon atoms of the predicted structure and reference, measuring local geometric accuracy. Details of the implementation are in Appendix A.

TM-score: Length-normalized global structural similarity between the predicted structure $\hat{\mathbf{x}}$ and reference $\mathbf{x}^*$. Unlike Cα-RMSD, TM-score is less sensitive to local deviations and allows $L \neq L^*$, making it suitable for comparing structures of sequences that diverge from the native.

3.2. Reward and feedback formulation

Global reward. For each candidate sequence, we predict its 3D structure using RosettaFold3 in single-sequence mode (no MSA conditioning), to avoid inflating confidence estimates via evolutionary priors from homolo-

gous sequences (Korbeld et al., 2025). This yields a stricter, sequence-intrinsic confidence measure. We compute the multi-objective reward as a weighted sum: $R(s) = w_{\text{tm}}\,\text{TM}(\hat{\mathbf{x}}, \mathbf{x}^*) + w_p\,\text{pLDDT}(\hat{\mathbf{x}}) + w_r\,C\alpha\text{-RMSD}(\hat{\mathbf{x}}, \mathbf{x}^*)$, scaled to $[0, 1]$, with $w_p = \frac{1}{11}$, $w_{\text{tm}} = w_r = \frac{5}{11}$. C$\alpha$-RMSD values exceeding a penalty threshold incur an explicit negative correction. A textual description of the reward quality accompanies the scalar value (e.g., "0.3 is a very low reward. Significant improvements needed.").

Local text feedback. Optimization using global information alone does not yield substantial improvements in preliminary experiments. We therefore additionally provide *local* textual feedback directing the LLM toward problematic regions. For pLDDT, low-confidence regions are detected via adaptive thresholds and reported as contiguous residue ranges. For Cα-RMSD, per-residue inter-atomic distances between $\hat{\mathbf{x}}$ and $\mathbf{x}^*$ are binned, and contiguous intervals for each bin are reported. This encourages localized sequence refinement rather than global rewrites. An example prompt with feedback is shown in Appendix Figure 13.

Multi-modal image feedback. When using VLMs, we additionally provide a rendered image of the predicted 3D structure as feedback. The structural image conveys global spatial context—secondary structure organization, domain topology, and exposed versus buried regions—that is difficult to convey through sequence or scalar metrics alone. We find this produces richer chain-of-thought reasoning, with the VLM referencing structural features such as helix packing and loop geometry when proposing edits.

Feedback to prevent reward hacking. Under optimization pressure, LLMs discover strategies that maximize proxy rewards while deviating from valid protein design—replicating short motifs with high pLDDT, producing sequences of incorrect length, appending poly-alanine runs, or returning the native sequence as a trivial local optimum. These resemble classic reward-hacking failure modes (Skalse et al., 2022). We introduce soft constraint feedback communicated in natural language: (i) a repetitiveness constraint flagging sequences where repeated 6-mers exceed 5% of total 6-mers; (ii) a length constraint flagging sequences deviating from the target length; and (iii) ligand-binding annotations identifying residues involved in functional interactions that must be retained.

3.3. Generative optimization with LLMs

We consider two iteration schemes for deploying generative optimizers.

Sequential revision. Our work adapts the LLM optimizer in Trace (Cheng et al., 2024) (a ReAct-style optimization agent), treating the protein sequence as a trainable text variable. At each iteration, the optimizer constructs a structured problem instance from intermediate states, outputs, and scalar or textual feedback. An LLM reasons over this representation and proposes targeted sequence updates. We run the optimizer iteratively to improve the starting sequence.

Priority search. Sequential optimization is prone to early convergence to local optima. As an alternative, we design a priority-based parallel search algorithm. Our search maintains a buffer storing all previous design attempts and their evaluation results, initialized with the starting protein sequence. At each iteration, we select the top-K performing candidates and evaluate them in parallel to obtain reward and feedback. We then generate K new candidates in parallel (one per candidate), evaluate them, and add all $2K$ candidates back to the buffer. See Algorithm 1 for the full procedure.

Multi-objective optimization. We explored several strategies for handling the multi-objective nature of protein design (see Appendix B). The weighted-sum formulation described above provided the most consistent optimization behavior and was adopted as the final configuration.

Step-size control. To regulate the magnitude of sequence modifications, we introduce a textual step-size mechanism encoding the number of permitted residue edits in the natural-language objective. We define five regimes from conservative (1–2 edits) to drastic (up to 20 edits), mirroring step-size selection in gradient-based optimization.

Best-of-N selection. For each protein design task, we run RosettaSearch under three step-size regimes (aggressive: 10 edits; moderate: 5 edits; unconstrained: any edits), each with a budget of 25 structure evaluations, for a total of 75 oracle calls per protein. The final output is the highest-reward candidate across all runs, analogous to best-of-N sampling in LLM decoding.

Final design choices. We use `o4-mini` as the LLM. For each optimization task, we run one Priority Search per step-size regime with $K = 3$ candidates per round and $N = 2$ proposals per candidate.

4. Datasets

PDB Monomers (LigandMPNN dataset). We curated a dataset from PDB-D (Xue et al., 2025), randomly sampling ≈400 monomeric structures with low fidelity metrics. For each structure, we utilize the native sequence and redesigned sequences from LigandMPNN (Dauparas et al., 2025), allowing us to assess whether the LLM can optimize

sequences to improve fidelity metrics while maintaining the target structure.

Dayhoff atlas. The Dayhoff BackboneRef dataset (Yang et al., 2025) consists of *de novo* protein backbones generated with RFDiffusion (Watson et al., 2023), with sequences designed by ProteinMPNN (Dauparas et al., 2022). These structures exhibit high baseline fidelity, providing a stringent testbed for evaluating whether optimization can improve well-folded designs. Crucially, no reference sequence is provided to the LLM in this setting—the optimizer acts solely on structural feedback.

BindCraft sequences. We evaluate on 50 protein binders from the BindCraft study (Pacesa et al., 2025), designed using a computationally intensive pipeline involving large-scale structure prediction and AlphaFold2 optimization. These represent already highly optimized candidates, providing a stringent test of inference-time refinement without access to the original pipeline.

5. Experimental Results

We evaluate RosettaSearch to answer: (i) can inference-time feedback-driven optimization reliably improve fidelity metrics and by how much; (ii) do gains persist under independent structure predictors and diverse initializations; (iii) what role does context play; (iv) does visual feedback from protein structure images help; and (v) does LLM reasoning carry useful information.

5.1. Evaluation metrics

We measure improvements in pLDDT, TM-score, and Cα-RMSD. We additionally evaluate using:

Success rate-1: Fraction of designs achieving pLDDT ≥ 0.8 *and* TM-score ≥ 0.8 simultaneously.

Success rate-2: Stricter criterion additionally requiring Cα-RMSD ≤ 1.5 Å.

Independent oracle evaluation. Evaluating with the same model used for optimization can lead to inflated estimates. We additionally evaluate all final designs using Chai-1 (Discovery et al., 2024), which differs in architecture, training data, and inductive biases from RosettaFold3.

5.2. RF3-guided random search baseline

To isolate the LLM's contribution from improvements due to random exploration, we introduce a baseline that is: (1) compute-matched—given the same RF3 oracle budget; (2) information-matched—provided the same residue-level feedback identifying low-pLDDT and low-TM-score regions; and (3) initialization-matched—starting from the same LigandMPNN designs. At each step it applies the same number of residue modifications as RosettaSearch, but makes random amino acid substitutions within the flagged regions. The best design is selected by best-of-N sampling.

Table 1. Sequential revision vs. Priority Search: absolute values and % improvement. Initialized from LigandMPNN designs. See Appendix Table 8 for statistical significance.

Single objective (pLDDT)			
	ss-pLDDT	TM-score	$C\alpha$-RMSD
Seq. Rev.	0.59→0.67	0.44→0.54	19.2→14.6
	17.0%	35.5%	−20.0%
Priority Search	0.59→0.69	0.44→0.57	19.2→13.0
	19.0%	**50.9%**	**−30.1%**
Multi-objective (ss-pLDDT, TM-score, $C\alpha$-RMSD)			
	ss-pLDDT	TM-score	$C\alpha$-RMSD
Seq. Rev.	0.59→0.71	0.44→0.61	19.2→11.7
	24.2%	65.1%	−32.6%
Priority Search	0.59→0.68	0.44→0.62	19.2→11.2
	18.0%	**67.7%**	**−37.7%**

As shown in Table 3, this baseline achieves no improvement over its LigandMPNN initialization—the highest-scoring candidate across runs is typically the original LigandMPNN design itself, indicating that random substitutions within flagged regions consistently degrade rather than improve structural fidelity. This demonstrates that residue-level feedback alone is insufficient: the LLM's ability to interpret that feedback and make targeted, chemically informed substitutions is the critical ingredient driving RosettaSearch's improvement.

5.3. Priority search outperforms sequential revision

Table 1 shows that priority-based parallel search consistently outperforms sequential revision. In sequential revision, each step greedily conditions on the current best sequence, leading to premature convergence when early edits introduce structural inconsistencies that later steps cannot recover from. Priority search explores multiple modification trajectories in parallel, deferring commitment to a single trajectory until downstream structural signals are observed. This yields higher success rates under identical query budgets.

5.4. RosettaSearch improves sequence fidelity

On ≈400 protein redesign examples, RosettaSearch improves ss-pLDDT by 18.1%, TM-score by 56.9%, and reduces Cα-RMSD by 31.2% using single-objective optimization (Table 1). Multi-objective optimization yields larger improvements overall. Notably, even single-objective optimization targeting only pLDDT produces substantial gains in TM-score and Cα-RMSD, reflecting the correlated nature

Table 2. Stringent success rate-2 (pLDDT≥0.8, TM-score≥0.8, Cα-RMSD≤1.5 Å).

	LigandMPNN	RosettaSearch
Success rate-2	2.5%	8.9%
Number of successes	(10/394)	(35/394)

of these structural metrics.

Table 3 shows success rates across different initialization settings. Starting from LigandMPNN designs, RosettaSearch achieves a **2.5× improvement** in design success rate (20.5% vs. 7.9% under RF3), corresponding to 79 vs. 32 successful designs out of 394 proteins. Gains persist under Chai-1 evaluation (≈2×: 6.7% vs. 3.1%), confirming that improvements are not artifacts of oracle overfitting.

Under the more stringent success rate-2 criterion (Table 2), RosettaSearch achieves 8.9% vs. LigandMPNN's 2.5%—a 3.5× improvement (35 vs. 10 successes out of 394).

When starting from random protein sequences with native monomer sequences available as context, RosettaSearch achieves a success rate of 8.1%, matching that of LigandMPNN, which directly conditions on the template structure.

Generalization across LLM families. Gemini-3 achieves 19.4% design success rate, comparable to o4-mini's 20.5%, demonstrating that gains are not specific to any single model family. Within the same family, o3-mini achieves 12.3%—substantially above the 7.9% LigandMPNN baseline but below o4-mini, consistent with its weaker reasoning capability. Performance scales with reasoning capability both across and within model families (Greisen et al.).

Sequence diversity. Figure 5 shows that although mean sequence identity between starting and generated sequences is 82.4%, the distribution is wide, indicating that the method adapts the extent of modification on a per-protein basis—making conservative edits when sufficient and exploring more aggressive redesigns when required.

Computational cost. Wall clock time per protein ranges from ≈2 min to ≈1.4 hours (average 30.6 min), dominated by RosettaFold3 inference.

5.5. Fidelity metrics improve over search steps

Figure 3 shows the averaged evolution of fidelity metrics and composite reward across 78 successful optimization trajectories. Monotonically filtered curves (dashed) show consistent improvement across all metrics, confirming that RosettaSearch performs meaningful search rather than benefiting from sampling luck. Unfiltered curves (solid) exhibit expected non-monotonic fluctuations reflecting the exploratory nature of priority search, while the overall trend remains consistently positive.

5.6. Multi-modal extension with vision-language models

As shown in Table 4, the VLM variant achieves a success rate of 16.6% (RF3) and 7.1% (Chai-1) when initialized from LigandMPNN designs, comparable to the text-only multi-objective variant (17.4% / 6.7%). Notably the VLM matches or exceeds the text-only variant on Chai-1, demonstrating that the multi-modal variant achieves performance parity with the text-only framework. While the raw metrics show marginal variance—with the VLM performing slightly lower on RF3 but showing a minor edge under independent Chai-1 evaluation—we consider this variant a vital proof-of-concept for spatial reasoning. Textual lists of low-confidence residue ranges fundamentally strip away three-dimensional spatial geometry. In contrast, visual feedback provides macro-structural context (e.g., secondary structure packing and solvent exposure) directly to the model's chain-of-thought, laying a framework for multi-modal protein optimization models

5.7. RosettaSearch improves BindCraft binders

Starting from 50 high-quality binders from BindCraft (Pacesa et al., 2025), RosettaSearch improves all 48 evaluated binders (Table 5): pLDDT $0.85 \rightarrow 0.90$, TM-score $0.93 \rightarrow 0.95$, Cα-RMSD $1.37 \rightarrow 0.99$ Å, while retaining binding pocket contacts with target enzymes. This demonstrates that inference-time search can refine already highly optimized designs without access to the original training pipeline.

While the absolute numerical increases in pLDDT ($0.85 \rightarrow 0.90$) and TM-score ($0.93 \rightarrow 0.95$) for the BindCraft subset appear incremental, they occur at the performance ceiling of an already heavily optimized pipeline. In this regime, minor shifts in confidence scores carry significant biological weight. Specifically, reducing the mean $C\alpha$-RMSD from 1.37 Å to a near-native 0.99 Å shifts the predicted ensembles into rigid, atomic-resolution stability. Crucially, sequence analysis confirms that RosettaSearch achieved these stabilization gains by introducing mutations primarily into the scaffold core rather than mutating critical interface residues.

5.8. Evaluation on the Dayhoff dataset

The Dayhoff subset consists of 275 randomly sampled RFDiffusion *de novo* backbones with ProteinMPNN-designed sequences that already have strong baseline fidelity. RosettaSearch improves success rate-1 from 72.4% to 89.5% and success rate-2 from 45.5% to 67.3% (Table 6). No ref-

Table 3. **(Upper)** Baseline success rates and RF3-guided random search baseline. **(Lower)** RosettaSearch results under different initialization and objective settings, evaluated by RF3 and independently by Chai-1.

		SR (RF3)	**SR (Chai-1)**
Baseline success rates			
(a) Native monomer sequences		7.0%	2.4%
(b) LigandMPNN designs		7.9%	3.1%
(c) Random protein sequences		0.0%	0.0%
RF3-guided random search (LigandMPNN init.)		7.9%	3.1%
RosettaSearch	**Objective**		
(a) Native monomer sequences	Single obj. (ss-pLDDT)	7.5%	4.2%
	Multi-obj. (ss-pLDDT, TM-score)	**9.1%**	**4.7%**
	Multi-obj. (ss-pLDDT, TM-score, Cα-RMSD)	8.4%	4.3%
	Cumulative	**10.1%**	—
(b) LigandMPNN designs	Single obj. (ss-pLDDT)	18.4%	5.0%
	Multi-obj. (ss-pLDDT, TM-score)	17.1%	**6.7%**
	Multi-obj. (ss-pLDDT, TM-score, Cα-RMSD)	**20.5%**	5.3%
	Cumulative	**21.6%**	—

Table 4. VLM results (Sequential Revision) vs. text-only LLM, optimizing ss-pLDDT and TM-score jointly.

			SR (RF3)	**SR (Chai-1)**
Init.	**Algorithm**	**Approach**		
(a) Native monomer sequences	Seq.-Rev.	Unoptimized native sequences	7.0%	2.4%
		LLM multi-obj. (ss-pLDDT, TM-score)	**8.9%**	**4.7%**
		VLM multi-obj. (ss-pLDDT, TM-score)	8.3%	4.6%
(b) LigandMPNN designs	Seq.-Rev.	Unoptimized LigandMPNN designs	7.9%	3.1%
		LLM multi-obj. (ss-pLDDT, TM-score)	**17.4%**	6.7%
		VLM multi-obj. (ss-pLDDT, TM-score)	16.6%	**7.1%**

Table 5. Improving BindCraft-designed binders.

Metric	BindCraft	RosettaSearch
ss-pLDDT ↑	0.85	0.90
TM-score ↑	0.93	0.95
Cα-RMSD ↓	1.37 Å	0.99 Å

Table 6. Results on Dayhoff *de novo* backbones (275 proteins).

Metric	ProteinMPNN	RosettaSearch
ss-pLDDT ↑	0.81	0.85
TM-score ↑	0.90	0.96
Cα-RMSD ↓	3.32 Å	1.78 Å
Success rate-1	72.4% (199/275)	89.5% (246/275)
Success rate-2	45.5% (125/275)	67.3% (185/275)

erence sequence is provided to the LLM in this setting—the optimizer must act solely on structural feedback from RosettaFold3. This makes the Dayhoff evaluation a stricter test, demonstrating that RosettaSearch does not rely on reference sequence context and generalizes beyond native protein structures to computationally generated backbones.

5.9. Ablation of contextual signals

Table 7 isolates the contribution of different context and feedback forms, starting from LigandMPNN designs. Reward-only sequential revision yields a minor improvement (8.5%→9.3%), indicating scalar rewards alone provide limited guidance. Adding detailed textual feedback improves performance to 11.4%. Providing the native reference sequence as context produces a large jump to 17.2%. Protein functional annotations add minimal further gain (17.4%), and RAG from a larger dataset slightly reduces performance (16.1%). These results motivate our design choice to provide at most the native sequence as context, which also limits input token length.

Table 7. Ablation of contextual signals (Sequential Revision, LigandMPNN initialization).

LigandMPNN designed sequences	8.5%
+ reward-only sequential revision	9.3%
+ detailed feedback text	11.4%
+ native sequence in context	17.2%
+ protein function in context	17.4%
+ RAG from bigger dataset	16.1%

5.10. Human and LLM expert analysis of reasoning

We examined LLM reasoning traces with 3 domain experts in biochemistry, structural biology, biophysics, and protein engineering (Figure 10). The LLM either proposes targeted, minimal edits focused on low-confidence regions or generates an entirely new sequence satisfying global design constraints. Experts noted the reasoning follows a fixed pattern of focusing on N- and C-terminal regions with low TM-scores—though these regions are typically flexible and loop-rich, where close structural agreement is not expected even for well-designed sequences. Sequence modifications were judged to make sense in isolation (e.g., replacing buried charged residues with nonpolar ones) but were sometimes inconsistent with the structural context.

We additionally used Gemini-3 as an independent evaluator of reasoning quality. Three Gemini-3 experts evaluated 50 reasoning cases generated by o4-mini, judging 28 instances (56%) as valid and 22 (44%) as invalid. Failures stemmed primarily from mutation budget violation (the LLM plans 10 substitutions but produces 15–30+) and hallucinated residue identities or positions. A subset of failures were no-op generations where the sequence was unchanged despite detailed mutation plans. These results suggest that RosettaSearch succeeds despite imperfect LLM reasoning, and that improving reasoning reliability is a promising avenue for further gains. Full analysis is in Appendix G.

6. Conclusion

RosettaSearch demonstrates that LLMs can serve as effective generative optimizers for protein sequence design, achieving systematic improvements over state-of-the-art single-pass methods through inference-time search guided by structural feedback—without any model retraining. The RF3-guided random search baseline confirms that LLM reasoning is the essential ingredient: identical oracle access and feedback without LLM guidance produces no improvement. Priority search outperforms sequential revision by maintaining a diverse candidate pool. Multi-modal VLM feedback yields qualitatively richer structural reasoning and comparable or better generalization under an independent oracle. Performance scales with reasoning capability across model families, suggesting a clear path forward as frontier models improve.

A current limitation is the focus on a single best-scoring sequence per backbone rather than a diverse ensemble. Diversity-aware reward formulations are an important direction for future work. The optimization trajectories generated by RosettaSearch also constitute rich preference data for continual adaptation of sequence design models through post-training.

Code and data availability. All code, data, and generated sequences will be released on GitHub upon publication.

References

Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Pritzel, A., Ronneberger, O., Willmore, L., Ballard, A. J., Bambrick, J., Bodenstein, S. W., Evans, D. A., Hung, C.-C., O'Neill, M., Reiman, D., Tunyasuvunakool, K., Wu, Z., Žemgulytė, A., Arvaniti, E., Beattie, C., Bertolli, O., Bridgland, A., Cherepanov, A., Congreve, M., Cowen-Rivers, A. I., Cowie, A., Figurnov, M., Fuchs, F. B., Gladman, H., Jain, R., Khan, Y. A., Low, C. M. R., Perlin, K., Potapenko, A., Savy, P., Singh, S., Stecula, A., Thillaisundaram, A., Tong, C., Yakneen, S., Zhong, E. D., Zielinski, M., Žídek, A., Bapst, V., Kohli, P., Jaderberg, M., Hassabis, D., and Jumper, J. M. Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature*, pp. 1–3, May 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07487-w.

Agrawal, L. A., Tan, S., Soylu, D., Ziems, N., Khare, R., Opsahl-Ong, K., Singhvi, A., Shandilya, H., Ryan, M. J., Jiang, M., et al. Gepa: Reflective prompt evolution can outperform reinforcement learning. *arXiv preprint arXiv:2507.19457*, 2025.

Anishchenko, I., Pellock, S. J., Chidyausiku, T. M., Ramelot, T. A., Ovchinnikov, S., Hao, J., Bafna, K., Norn, C., Kang, A., Bera, A. K., et al. De novo protein design by deep network hallucination. *Nature*, 600(7889):547–552, 2021.

Cheng, C.-A., Nie, A., and Swaminathan, A. Trace is the next autodiff: Generative optimization with rich feedback, execution traces, and llms. *Advances in Neural Information Processing Systems*, 37:71596–71642, 2024.

Corley, N., Mathis, S., Krishna, R., Bauer, M. S., Thompson, T. R., Ahern, W., Kazman, M. W., Brent, R. I., Didi, K., Kubaney, A., et al. Accelerating biomolecular modeling with atomworks and rf3. *bioRxiv*, 2025.

Dauparas, J., Anishchenko, I., Bennett, N., Bai, H., Ragotte, R. J., Milles, L. F., Wicky, B. I. M., Courbet, A., de Haas,

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

Dauparas, J., Lee, G. R., Pecoraro, R., An, L., Anishchenko, I., Glasscock, C., and Baker, D. Atomic context-conditioned protein sequence design using LigandMPNN. *Nature Methods*, 22(4):717–723, April 2025. ISSN 1548-7105. doi: 10.1038/s41592-025-02626-1.

Discovery, C., Boitreaud, J., Dent, J., McPartlon, M., Meier, J., Reis, V., Rogozhnikov, A., and Wu, K. Chai-1: Decoding the molecular interactions of life, October 2024.

Gottweis, J., Weng, W.-H., Daryin, A., Tu, T., Palepu, A., Sirkovic, P., Myaskovsky, A., Weissenberger, F., Rong, K., Tanno, R., et al. Towards an ai co-scientist. *arXiv preprint arXiv:2502.18864*, 2025.

Greisen, A., Cong, L., Ovchinnikov, S., et al. Reasoning models outperform standard language models in de novo protein design. In *Open Conference of AI Agents for Science 2025*.

Khattab, O., Singhvi, A., Maheshwari, P., Zhang, Z., Santhanam, K., Vardhamanan, S., Haq, S., Sharma, A., Joshi, T. T., Moazam, H., et al. Dspy: Compiling declarative language model calls into self-improving pipelines. *arXiv preprint arXiv:2310.03714*, 2023.

Korbeld, K. T., Viliuga, V., and Fürst, M. Limitations of the refolding pipeline for de novo protein design. *bioRxiv*, pp. 2025–12, 2025.

Lutz, I. D., Wang, S., Norn, C., Courbet, A., Borst, A. J., Zhao, Y. T., Dosey, A., Cao, L., Xu, J., Leaf, E. M., et al. Top-down design of protein architectures with reinforcement learning. *Science*, 380(6642):266–273, 2023.

Nie, A., Cheng, C.-A., Kolobov, A., and Swaminathan, A. The importance of directional feedback for llm-based optimizers. *arXiv preprint arXiv:2405.16434*, 2024.

Pacesa, M., Nickel, L., Schellhaas, C., Schmidt, J., Pyatova, E., Kissling, L., Barendse, P., Choudhury, J., Kapoor, S., Alcaraz-Serna, A., Cho, Y., Ghamary, K. H., Vinué, L., Yachnin, B. J., Wollacott, A. M., Buckley, S., Westphal, A. H., Lindhoud, S., Georgeon, S., Goverde, C. A., Hatzopoulos, G. N., Gönczy, P., Muller, Y. D., Schwank, G., Swarts, D. C., Vecchio, A. J., Schneider, B. L., Ovchinnikov, S., and Correia, B. E. 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.

Pryzant, R., Iter, D., Li, J., Lee, Y. T., Zhu, C., and Zeng, M. Automatic prompt optimization with" gradient descent" and beam search. *arXiv preprint arXiv:2305.03495*, 2023.

Romera-Paredes, B., Barekatain, M., Novikov, A., Balog, M., Kumar, M. P., Dupont, E., Ruiz, F. J., Ellenberg, J. S., Wang, P., Fawzi, O., et al. Mathematical discoveries from program search with large language models. *Nature*, 625 (7995):468–475, 2024.

Skalse, J., Howe, N., Krasheninnikov, D., and Krueger, D. Defining and characterizing reward gaming. *Advances in Neural Information Processing Systems*, 35:9460–9471, 2022.

Stracquadanio, G. and Nicosia, G. Computational energy-based redesign of robust proteins. *Computers & chemical engineering*, 35(3):464–473, 2011.

Sun, H., Liu, X., Gong, Y., Zhang, Y., Jiang, D., Yang, L., and Duan, N. Allies: Prompting large language model with beam search. *arXiv preprint arXiv:2305.14766*, 2023.

Swanson, K., Wu, W., Nash, L. B., Pak, J. E., and Zou, J. The virtual lab: Ai agents design new sars-cov-2 nanobodies with experimental validation. biorxiv. 2024.

Wang, X., Li, C., Wang, Z., Bai, F., Luo, H., Zhang, J., Jojic, N., Xing, E. P., and Hu, Z. Promptagent: Strategic planning with language models enables expert-level prompt optimization. *arXiv preprint arXiv:2310.16427*, 2023.

Watson, J. L., Juergens, D., Bennett, N. R., Trippe, B. L., Yim, J., Eisenach, H. E., Ahern, W., Borst, A. J., Ragotte, R. J., Milles, L. F., Wicky, B. I. M., Hanikel, N., Pellock, S. J., Courbet, A., Sheffler, W., Wang, J., Venkatesh, P., Sappington, I., Torres, S. V., Lauko, A., De Bortoli, V., Mathieu, E., Ovchinnikov, S., Barzilay, R., Jaakkola, T. S., DiMaio, F., Baek, M., and Baker, D. De novo design of protein structure and function with RFdiffusion. *Nature*, pp. 1–3, July 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-06415-8.

Xia, Y., Jin, P., Xie, S., He, L., Cao, C., Luo, R., Liu, G., Wang, Y., Liu, Z., Chen, Y.-J., et al. Nature language model: Deciphering the language of nature for scientific discovery. *arXiv preprint arXiv:2502.07527*, 2025.

Xue, F., Kubaney, A., Guo, Z., Min, J. K., Liu, G., Yang, Y., and Baker, D. Improving Protein Sequence Design through Designability Preference Optimization, May 2025.

Yang, K. K., Wu, Z., and Arnold, F. H. Machine-learning-guided directed evolution for protein engineering. *Nature methods*, 16(8):687–694, 2019.

Yang, K. K., Alamdari, S., Lee, A. J., Kaymak-Loveless, K., Char, S., Brixi, G., Domingo-Enrich, C., Wang, C., Lyu, S., Fusi, N., et al. The dayhoff atlas: scaling sequence diversity for improved protein generation. *bioRxiv*, pp. 2025–07, 2025.

Yin, R., Feng, B. Y., Varshney, A., and Pierce, B. G. Benchmarking alphafold for protein complex modeling reveals accuracy determinants. *Protein Science*, 31(8):e4379, 2022.

Yuksekgonul, M., Bianchi, F., Boen, J., Liu, S., Huang, Z., Guestrin, C., and Zou, J. Textgrad: Automatic" differentiation" via text. *arXiv preprint arXiv:2406.07496*, 2024.

Zhang, Y. and Skolnick, J. Tm-align: a protein structure alignment algorithm based on the tm-score. *Nucleic acids research*, 33(7):2302–2309, 2005.

Zhou, X., Xue, D., Chen, R., Zheng, Z., Wang, L., and Gu, Q. Antigen-specific antibody design via direct energy-based preference optimization. *Advances in Neural Information Processing Systems*, 37:120861–120891, 2024.

Appendix

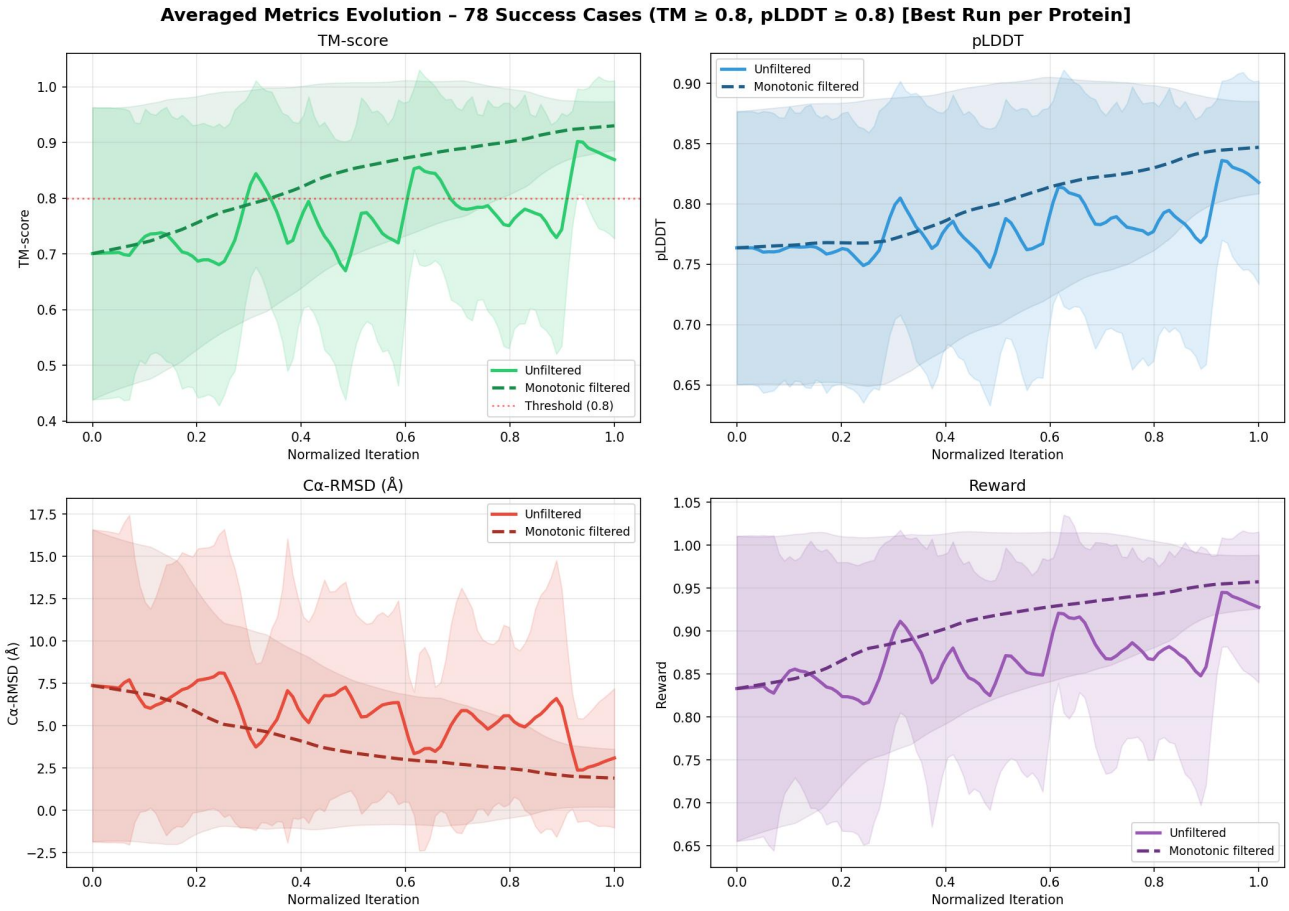


Figure 3. Averaged metric and reward evolution across 78 successful cases (TM-score ≥ 0.8, pLDDT ≥ 0.8) over normalized optimization iterations. Solid lines show unfiltered per-iteration values; dashed lines show monotonically filtered trajectories. Shaded regions show standard deviation across proteins. Results from Priority Search with multi-objective optimization.

A. Calculation of Cα-RMSD

To measure $C\alpha$-RMSD$(\hat{\mathbf{x}}, \mathbf{x}^*)$ where $L = L^*$, we superimpose the predicted structure over the reference using the TM-align algorithm (Zhang & Skolnick, 2005), which aligns remotely homologous structures (sequence identity <30%). We use this approach since the generated sequence may have arbitrarily low sequence similarity to the native.

B. Multi-objective optimization details

We evaluated two alternative multi-objective strategies in addition to the weighted-sum formulation adopted in the main paper.

First, a **round-robin scheme** alternating between objectives across iterations, allowing the model to focus on a single criterion at a time while implicitly encouraging coverage of the full objective set. Candidate evaluations are ranked using a single objective at a time, with the active objective cycling across searches.

Second, **scheduled reward formulations** in which the relative emphasis on different objectives varies over the course of optimization, including curricula that prioritize coarse global structure in early iterations and increasingly emphasize local refinement signals in later iterations.

The weighted-sum formulation provided the most consistent optimization behavior across our experiments and was therefore

adopted as the final configuration.

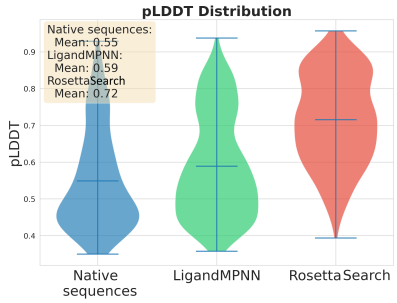
C. Statistical significance

Table 8. Sequential revision vs. Priority Search, mean % improvement with standard error and paired t-test p-values.

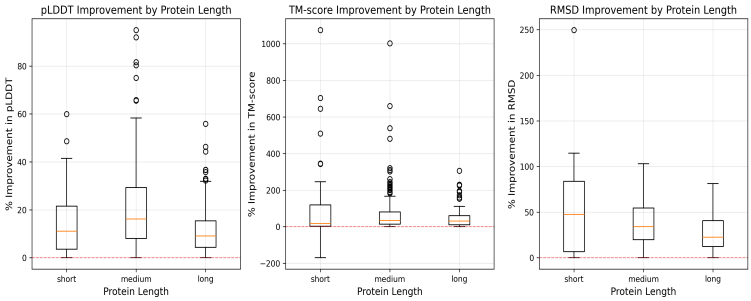
Single objective (pLDDT)			
	ss-pLDDT	TM-score	$C\alpha$-RMSD
Sequential Revision	19.4%	27.4%	−15.0%
Priority Search	18.1±0.85%	56.9±4.92%	−31.2±1.28%
	p=1.61×10^{-77}	p=3.42×10^{-57}	p=8.71×10^{-40}
Multi-objective (ss-pLDDT, TM-score, $C\alpha$-RMSD)			
	ss-pLDDT	TM-score	$C\alpha$-RMSD
Sequential Revision	24.2±1.1%	65.1±5.2%	−32.6±1.3%
	p=1.9×10^{-81}	p=2.1×10^{-65}	p=1.42×10^{-40}
Priority Search	18.0±0.82%	67.7±5.82%	−37.7±1.24%
	p=4.9×10^{-81}	p=6.1×10^{-68}	p=1.1×10^{-45}
Cumulative	38.5%	86.4%	−44.4%

D. Results by protein length

Results stratifying % improvement in fidelity metrics by protein length are shown in Figure 4b, where *short* refers to sequences with fewer than 100 amino acids, *medium* to 100–350 amino acids, and *long* to more than 350. Our PDB dataset has 249 medium, 129 long, and 36 short monomers. Improvements in pLDDT and TM-score are highest for medium-length proteins, while Cα-RMSD improvements are highest for short proteins.



(a) Distribution of pLDDT values for LigandMPNN baseline designs and RosettaSearch-optimized designs (single-objective pLDDT optimization).



(b) % improvement stratified by sequence length. (*Left*) pLDDT, (*Center*) TM-score, (*Right*) Cα-RMSD. Multi-objective optimization.

Figure 4. Metric distributions and length-stratified improvements.

E. Where are the most updates made?

To understand LLM substitution propensity, we divide each sequence into four bins: N-terminal (0–25%), two middle bins, and C-terminal (75–100%). Comparing successful generated sequences to starting sequences, the C-terminal sees the highest percentage of updates (30%), while other regions receive 22–24% of updates (Figure 6).

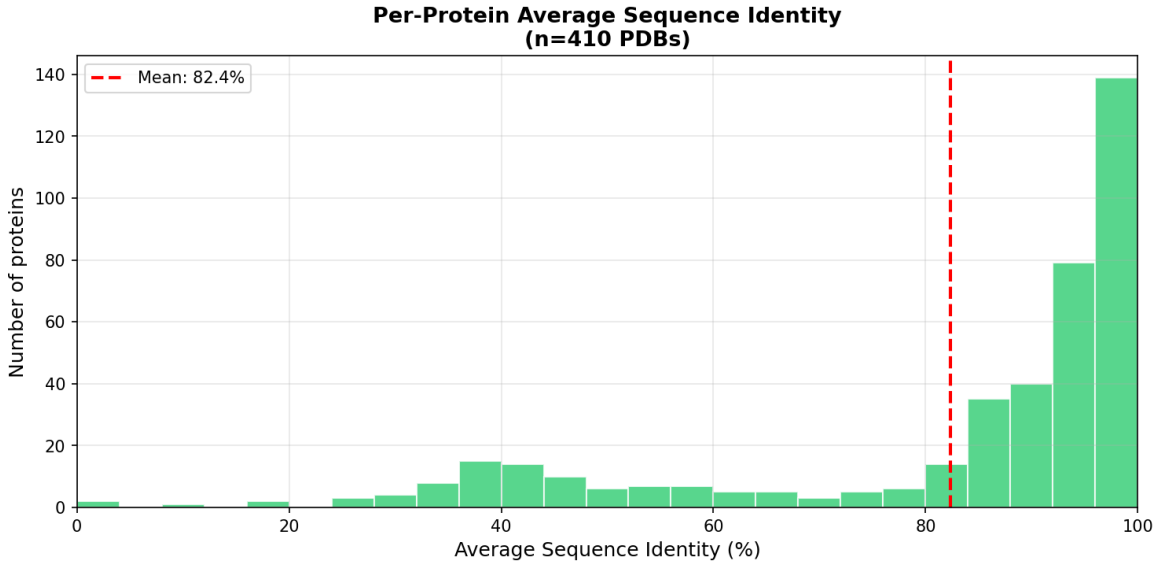


Figure 5. Distribution of sequence identity between starting and generated sequences on the LigandMPNN dataset (400 monomers). Mean identity is 82.4% but the distribution is wide, indicating per-protein adaptation.

F. Amino-acid substitution patterns

We compute residue-level substitution matrices recording how frequently each amino acid in the reference is replaced by another in the designed sequence. LigandMPNN exhibits more stereotyped, high-probability substitutions consistent with learned inverse-folding priors (Figure 7), whereas RosettaSearch shows more local exploration with some overuse of structurally safe residues (Figure 8).

G. Hallucination analysis

We analyzed the 22 reasoning cases deemed invalid by Gemini-3 expert evaluation. Failures overwhelmingly stem from reasoning–action inconsistencies rather than poor design intent. The dominant failure mode is mutation budget violation: the LLM explicitly plans at most 10 substitutions but produces sequences with 15–30+ mutations. A second class involves hallucinated inputs or indexing mistakes—incorrect assumptions about the native sequence, misidentified residue identities or positions, and reasoning about regions not present in the feedback. A subset of failures are no-op generations where the proposed sequence is unchanged despite detailed mutation plans.

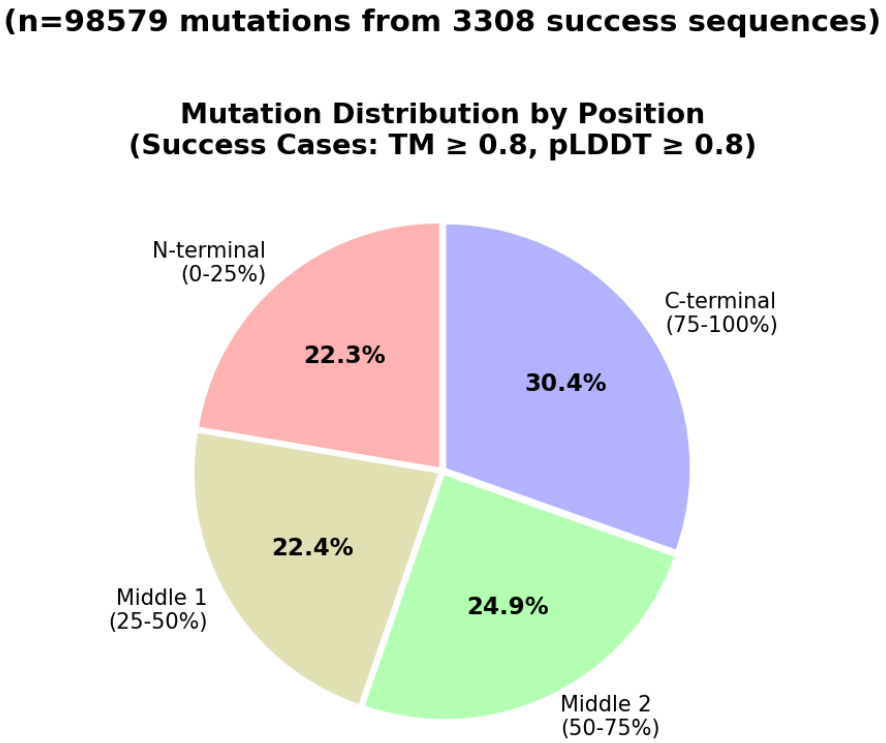


Figure 6. Distribution of updated positions in generated vs. starting sequences. 3,308 generated sequences satisfied the success criterion across 400 proteins' trajectories.

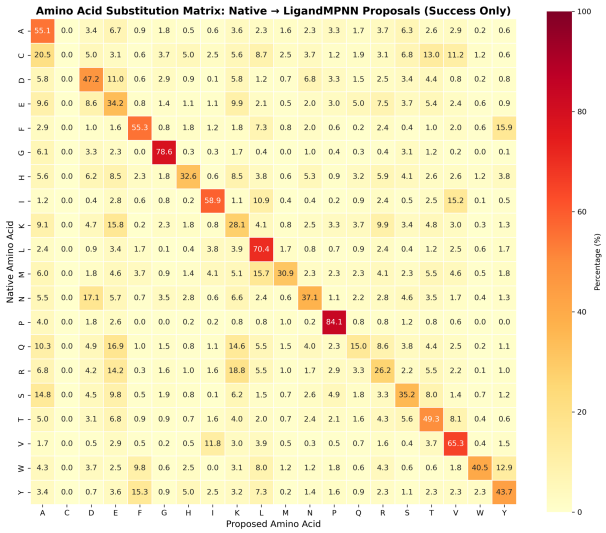


Figure 7. Substitution matrix: native → LigandMPNN designs.

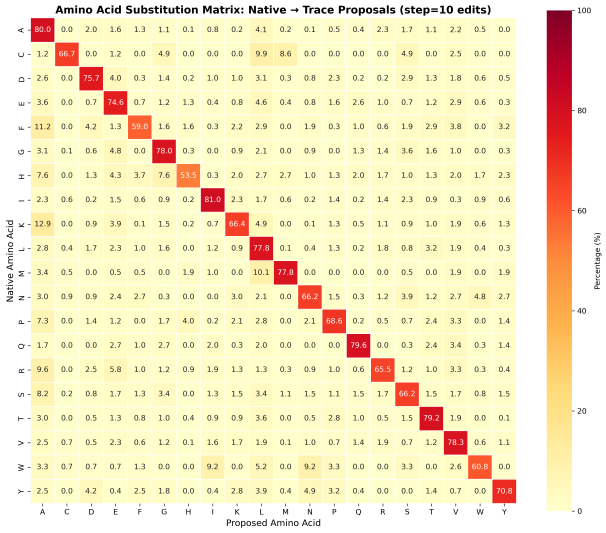


Figure 8. Substitution matrix: native → RosettaSearch designs (native sequence initialization).

H. RF3 vs. Chai-1 evaluation

I. Expert annotation interface

J. Example structures

K. Example prompt

L. Example sequences

M. Priority Search algorithm

Algorithm 1 Priority Search

Require: ϕ_0: starting protein sequence
$\mathcal{O}$: LLM Optimizer
G: Guide providing feedback (pLDDT, TM-score, Cα-RMSD)
K: number of candidates for exploration
N: number of proposals per candidate
$S(\cdot)$: score function to prioritize candidates

Initialize priority queue $\mathcal{M}$ storing candidates $c = (\phi, \text{score})$
Insert K copies of ϕ_0 into $\mathcal{M}$ with maximal priority
for iteration $t = 1, 2, \ldots$ **do**
  (Exploit) $c^* \leftarrow \arg\max_{c \in \mathcal{M}} S(c)$

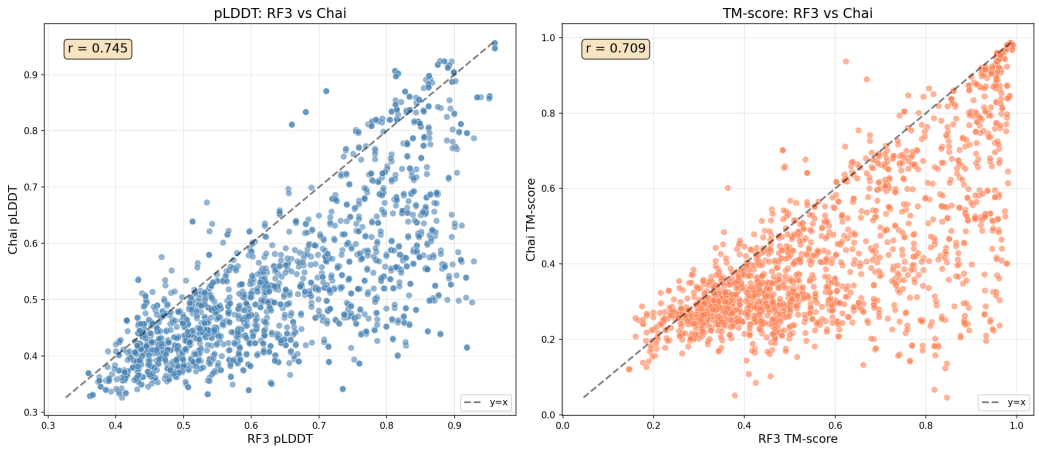


Figure 9. Comparison of pLDDT and TM-score from RF3 and Chai-1 for sequences generated by RosettaSearch (multi-objective). Each point represents one generated sequence. Chai-1 tends toward more conservative predictions.

Dayhoff id	ProteinMPNN design	RosettaSearch optimized	Seq. id.
20240891510_CvNknz_00140AA_2	AAEKEKEELKKAREKMEKEIKE LVEKLKGYIIISILTAVLREVSGK LVVEIKVVAGSESIKLEELEAKM KEYIAKKKAEFAALGLKVEEEIK RSEKEVVAKLTAPLDKLPPESD KLKRTLEIINLETGEVEVRKEEIT LE	MKAKQKLALKKAREKMALQIK ELVAKLKGYIIISILTAVLREVTA KLVVEIKVVAGSASIRLEELEAK MKEYIAKKKAEFAALGLKVEEE IKPSPEEVEVKLTAPLDKLPESD TLKRTLEIINLETGEVEVRKEEIT LAA	88.2%
20240891627_ZY18lJ_00304AA_2	ATLELTGPLTPEFLAKARELLKP LAAAEEALTALDEVVALGNATV TIYKQQGWTTTSTLTLTIERRPN EEKVTVTLTVTATAADGTTRTT TITATVTYVELPGGPPVLKRTLE EKASDGSSFTLSISTTGAMELSL QVPGKEPVSLTLDIIKGGELERK FYVDPSSGNRIIVIKATKDHIVVI IRIRKLTPESKELLKEILEEVFKE AKKEGAKVHVVILADTLENAKI ALDIVLEQALKEGVTLSISLGVA VKAEEADEVYEELVKYVEEKLE KIREEGKVKVERVEVRLPGKTA VFEVEEEI	MTLENTAALTAEFLAKARELLK PLAAAEEALTALDEVVALGLAT VTILALQGWTTTSTLTLTIERKA NEEKVTVTLTLTALAANGTTRT TTITATVTYVELPGGPPVLKRTL EEKASDGSSFTLSISTTGAMELS LQVPGKEPVSLTLDIITGGELER KFYVDPSSGNRIIVIKATKDHIV VIIRIRKLTPESKELLKEILEEVFK EAKKEGAKVHVVILADTLENAK IALDIVLEQALKEGVTLSISLGV AVKAEEADEVYEELVKYVEEKL EKIREEGKVKVERVEVRLPGKT AVFEVEEEI	95.4%
20240891538_B8FFVA_00355AA_0	TAQLQADLDRVVAALAANPRIQ AIRAILAQRPEVEPEAVSNERAL ELVMEVLGLRDRAEARALMVP AARLFGVPVPPLSPEEQELLAAA LARPVLTVALVFRVRYSRFEVL EEEEREYEDPERPGLRVRERRRV VRVDGYLYRFVHIEVRDEATGK VVRAGLLSVQVGPDHYVISVDF EAEPGASEEAVDLVVDRVNEEL REIAKEIKKATTLSLSIRDTAHAP KVIQAFVAAANEIAPYVKSLKV SIHVGKHPGPITLTLTLTPGVDA LTVIAENCAELNVNVTAGVKEL RFTAYSTKITVTLAPDAKIEKVE FVETTPETVINTIAEVVLECLATE EDALGVVVLREEVGAVI	TAQLQADLDAVVAALAANPEIQ AIRAILAQRPEVLPEAVSNERAL ELVMIVLGLRDRAEARALMVPA ARLFGVPVPPLSPEEQELLAAAL ARPVLTVALVFRVRYSRFEVLE EEEREYEDPERPGLRVRERRRV VRVDGYLYRFVHIEVRDEATGK VVRAGLLSVQVGPDHYVISVDF EAEPGASEEAVDLVVDRVNEEL REIAKEIKKATTLSLSIRDTAHAP KVIQAFVAAANEIAPYVKSLKV SIHVGKHPGPITLTLTLTPGVDA LTVIAENCAELNVNVTAGVKEL RFTAYSTKITVTLAPDAKIEKVE FVETTPETVINTIAEVVLECLATE EDALGVVVLREEVGAVI	98.9%

Table 9. Dayhoff ProteinMPNN sequences redesigned by RosettaSearch (see Figure 12).

Example 1: 7v9h

Sequence Comparison

Mutations (12): A2P, L4K, A5S, A6G, K7S, E8D, Q50A, Y51E, N52A, S54K, P55A, S56A

Query length: 102 | **New length:** 102

```
Pos            10        20        30        40        50        60        70        80
Query MAELAAKECLLSKEQLRSIYESSLSIGNFASRLLVHLFPELFTHENLRKQYNASPSLGKKQLDPARIRLIRHYVQLLYPR
      | |     |||||||||||||||||||||||||||||||||||||||||||   |   |||||||||||||||||||||||
New   MPEKSGSDCLLSKEQLRSIYESSLSIGNFASRLLVHLFPELFTHENLRKAEAAKAALGKKQLDPARIRLIRHYVQLLYPR

Pos            90        0
Query AKNDRVWTLEFVGKLDERCRRR
      ||||||||||||||||||||||
New   AKNDRVWTLEFVGKLDERCRRR
```

LLM Reasoning:

The instruction asks us to improve the designability (structural similarity and confidence) by mutating at most 10 residues per step. The TM-score feedback highlighted poor structural match in residues 1–1, 3–4 and 50–58, and pLDDT flagged residue 3. Therefore we targeted those positions for mutation, choosing helix-favouring or stabilizing residues: we mutated position 1 (D→M) to re-introduce a start methionine and improve helix capping; positions 3–4 (R,N→E,K) to increase helical propensity and put a charged pair for intra-helix salt bridging; and positions 50–56 (Q,Y,N,C,S,G,S→A,E,A,A,K,A,A) to replace a poorly structured loop with an Ala/Glu-rich helix-favouring stretch. This is a total of 10 mutations. We expect better local secondary-structure formation in these regions, higher local TM-score and pLDDT improvements around residue 3 and the 50–58 segment.

3D Structures

Query structure vs New structure

Query Structure

New Structure (after LLM mutations)

Figure 10. Expert annotation interface showing LLM reasoning for one step of sequence optimization (randomly chosen example).

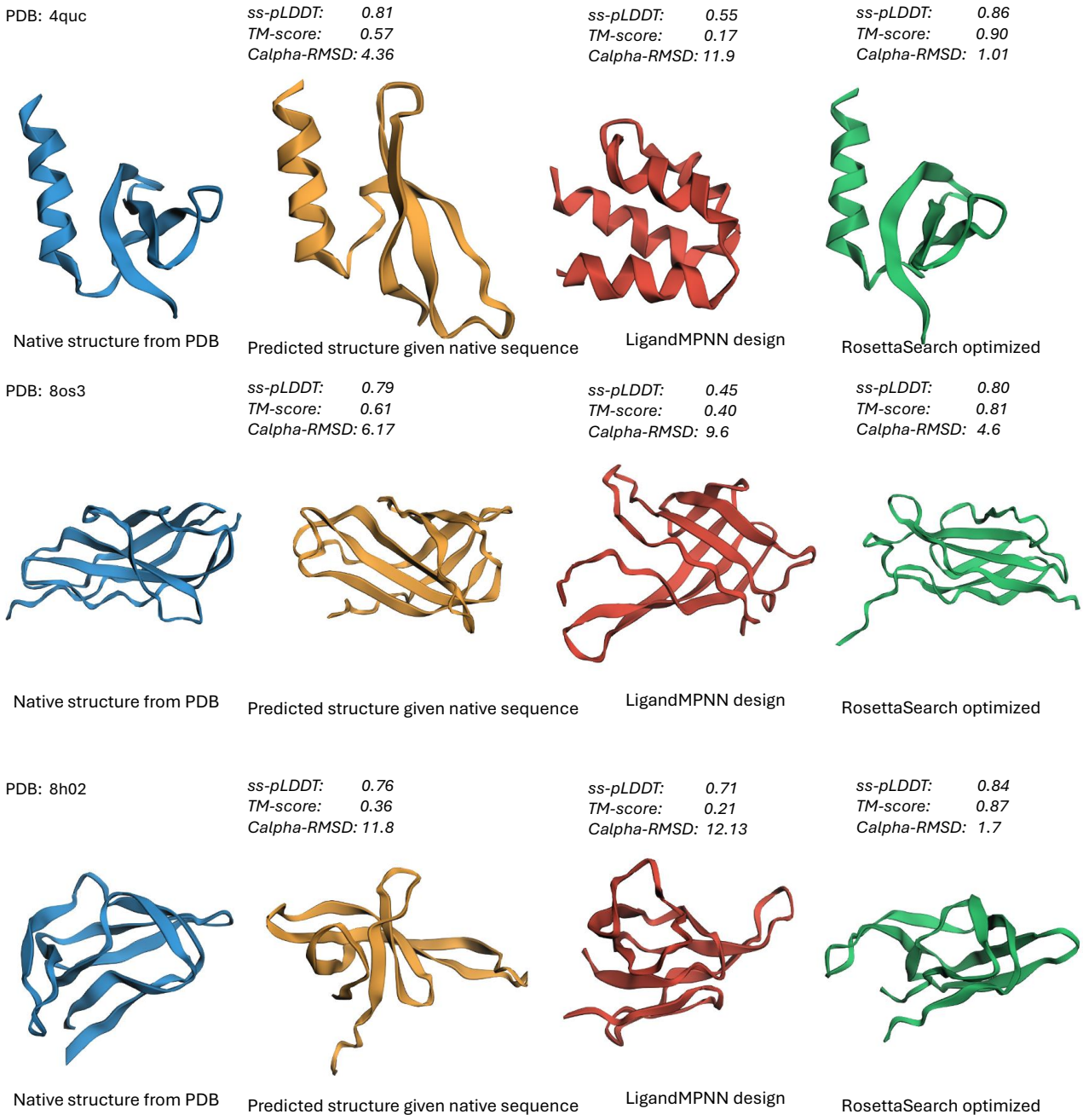


Figure 11. Additional PDB dataset structures successfully optimized by RosettaSearch. See Table 10 for sequences.

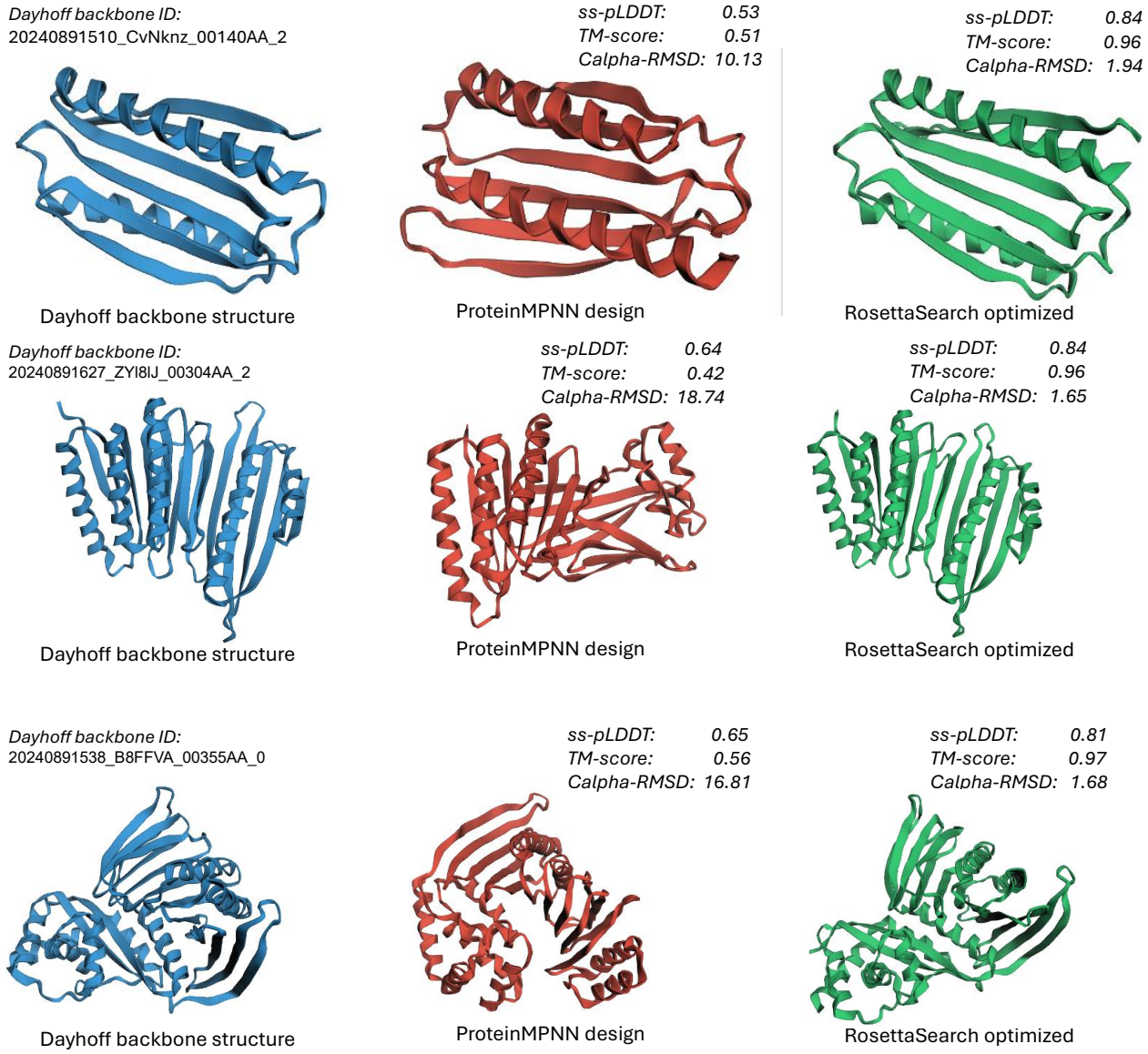


Figure 12. Dayhoff atlas structures successfully optimized by RosettaSearch. See Table 9 for sequences.

Figure 13. Example prompt input to the LLM for protein sequence optimization.

Example Optimization Prompt

```
#Instruction
We want to increase the structural fidelity of the proposed protein sequence, by
changing a maximum of 20 residues in each step.  The constraint is to match the fold
of this native protein sequence:
KSKLAKPIQDLIKMIFDVESMKKAMVEFEIDLQKMPLGKLSKRQIQSAYSILNEVQQAVSDGGSESQILDLSNRFYTLIPHDFGMK
KPPLLSNLEYIQAKVQMLDNLLDIEVAYSLLRGGNEDGDKDPIDINYEKLRTDIKVVDKDSEEAKIIKQYVKNTHAATHNAYDLKV
VEIFRIEREGESQRYKPFKQLHNRQLLWHGSRTTNFAGILSQGLRIAPPEAPVTGYMFGKGIYFADMVSKSANYCHTSQADPIGLI
LLGEVALGNMYELKNASHITKLPKGKHSVKGLGKTAPDPTATTTLDGVEVPLGNGISTGINDTCLLYNEYIVYDVAQVNLKYLLKL
KFNYKTS.
Use only standard amino acids (ACDEFGHIKLMNPQRSTVWY), no gaps.  Output only the raw
sequence without any code.

#Variables
(str) protein_sequence= MSKLAPPIVELIKDIFDRERLEEELRRMQIDLERLPLDSLSEEMIDRALRILTEVSRVVKNGG
SEEEIERLSEEFYALLPRDFGDEEPPLLDDPELIRQRVLDLDHLRRIRTTWRLLNGGLDSKEKHPIDILYEKLRTRIEVVDPDSEE
AKLIRKYVENTWGPDDNDFDIEVEEIFRIEREGAAERFAPYLKKENRWLLFHGTRITELAGILRDGLTLPPPEAPIEGYMFGKGIE
FTSVVSKAAKNMETSRENPIGYLLLAEVALGKMYKIKKPENIKELPEGYDSVHGIGKYAPDPKETTTLDGVKVPLGKLVETGETDT
FFKYDRFVVYDEAQVLLRYLLKLKFNFYGD

#Constraints
(str) A protein sequence of 351 amino acid residues.  Use only standard amino
acids:  ACDEFGHIKLMNPQRSTVWY. Do not include repeated patterns or gaps or motifs.
Output ONLY the raw sequence without any code, or formatting.  Example format:
MKTAYIAKQRQISFVK...

#Feedback
TM-score feedback:  The TM-score is low, indicating poor structural similarity.  The
following residue regions have low structural similarity (local average deviation
>5.0Å): 1-29, 32-41, 44-44, 46-53, 63-64, 72-74, 78-125, 140-157, 161-212, 217-267,
271-286, 290-308.  Focus on improving these regions.  pLDDT feedback:  The following
ranges have low pLDDT (<35):  111:111, 121:121, 137:137, 139:139, 180:180, 196:201,
204:206, 212:214, 216:222, 224:226, 228:237, 239:239, 247:247, 251:252, 255:255,
257:257, 262:263, 265:265, 267:271, 287:288, 290:290, 292:293, 318:333, 335:335,
339:340, 345:345, 351:351.  Redundancy feedback:  This generated sequence is
sufficiently non-redundant.  Low reward.  Significant improvements necessary.
```

PDB id	Native sequence	LigandMPNN design	RosettaSearch optimized
7z05	KESLTHASVLSAYTASYKTTAIKSIA DNAVVLDIGYGKGNDGPRYAVRPL TVTGIDTAARMLAIADQNKPENVTL VKQGFFTHITKTSNTYTHVIAFNSL HYPLASSHPDTLVQRLPTCPANILIP CHHLLEGIQTPTYSVVKDEDMWCV KVTKNEFIESSYNYDVFVKALESKY HVTIGSLLDCVEKPSTRSITPTLWTA MRNFVNNDQEMQRILSGYITFNLTP LPPKVEIINDWLD	EDGLTHADVLFAFRGSAQTRAVDA IPEGSTVLDIGVGTGRLFPLYARRPL TVTGIDVDAAALAKAEAHRPANVT LVHQDLFTYLKETDVEYTYVLAFD ALHYPLSRSDPRELVDLLPRAPAFIL IPDFRLLEGVETPTFSVRPDGDRVL VRIGGRELTYDYYDLDAFEAALAE RYDVRRGTLLDETTKPADPSITDEL WERMRALVANDEDLQRILGGFRTY DLTPKPPPAAAAAAAAA	MESLTHADVLFAYTASYKTTAIDAI PEGSTVLDIGVGTGRLFPLYARRPL TVTGIDVDAAALAKAEAHRPANVT LVHQDLFTYLKETDVEYTYVLAFD ALHYPLSRSDPRELVDLLPRAPAFIL IPDFRLLEGVETPTFSVRPDGDRVL VRIGGRELTYDYYDLDAFEAALAE RYDVRRGTLLDETTKPADPSITDEL WERMRALVANDEDLQRILGGFRTY DLTPKPPPAEIINDWLD
4quc	EYVVEKILGKRFVNGRPQVLVKWS GFPNENNTWEPLENVGNCMKLVSD FESEVFRL	SADPDKILGRLLRNGFPYYHIHWLH MTQENITWRSAAHLGNIGFQWILFR YALQRK	EYVVEKILGKRFVNGRPYYHIKWL GMTQENITWRSAANVVGCMKLVS DFESELQRK
8os3	GSSSPPSPPLDLHVTDAGRKHIAIA WKPPEKNGGSPIIGYHVEMCPVGTE KWMRVNSRPIKDLKFKVEEGVVPD KEYVLRVRAVNAIGVSEPSEISENV VAKD	GEVAAPTEPTSLKVTAASKDYIVIE WQSPESDGGTPKRGYSVQMVRVFT ENWVEVNSALIKSNSYQVTAGVTP NEVFKLRVTAFNDAGSSEPSQESEN VKKAT	GSAAAPTEPTSLKVTAASKDYIVIE WQSPEKNGGSPILGFSVQMVRVFTE NWVEVNSRPIKSNSYQVTAGVTPN EVFKLRVTALDQAGSAEPSEESENV VKAT
8h02	TARLLRAPVAGTIKLGKKARTRPY RTRHGEEALLAEANFDLVLEGKGR KETFAILQGSTIFVQDGDKVAAEAI LAEVPV	DYEVIFAHEPGFKILGDEAVLERYV TKDGNDVLRAKKNFTLIIVGKKKK HESNGPEGSGVLVRDGDTVDKGEP LAIVPL	TARLLRAPVAGTIILGDEAVLERYV TKDGNDVLRAKKNFTLIIVGKGRK ETFNGPEGSGVLVRDGDKVAAGEP LAIVPL

Table 10. LigandMPNN sequences redesigned by RosettaSearch, with native PDB sequences. Corresponds to Figures 1 and 11.