STENCIL: ALIGNMENT FIDELITY OUTWEIGHS TEMPLATE SELECTION IN RETRIEVAL-AUGMENTED PROTEIN INVERSE FOLDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Retrieval-augmented inverse folding conditions design on similar templates, and recent work invests in retrieving more and better ones. Within the template-retrieval setting studied here, candidate discovery already provides informative structural evidence. We therefore focus on the next post-retrieval bottleneck: conditional on an admissible template, can its residues be registered precisely at the target positions? STENCIL addresses this question by exhaustively re-aligning retained candidates by exact structural superposition, using a single template, and decoding in one deterministic pass with 3.2M parameters. On split-safe CATH 4.2 it reaches 69.342% residue-pooled recovery, the highest among methods with runnable deterministic re-evaluations under our protocol. The decisive test leaves the template, its coverage, its TM-score and its composition untouched, and permutes only where its residues land; that alone costs 26.95 recovery points. A complete-case paired 2×2 analysis finds that the alignment contrast is statistically larger than the selection contrast. The sequence-identity analysis bounds the regime in which retrieval helps.

1 INTRODUCTION

Inverse folding designs amino-acid sequences for prescribed backbones and is a core primitive of computational protein engineering (Ingraham et al., 2019; Dauparas et al., 2022). Retrieval-augmented methods extend it by supplying sequences from structurally similar templates as additional evidence (Liao et al., 2025; Mahbub et al., 2026; Zhu et al., 2026).

Recent work improves retrieval by expanding template pools, increasing the number of conditioning templates, and modelling template quality. Within the template-retrieval setting studied here, candidate discovery already provides informative structural evidence. We therefore focus on the next post-retrieval bottleneck: conditional on an admissible template, can its residues be registered precisely at the target positions?

Template similarity determines what evidence is available; registration fidelity determines how much of that evidence can be converted into sequence recovery. A stencil transfers a pattern only when it is registered exactly; shifting it by one notch destroys the pattern almost as completely as discarding it. We find that structural templates behave the same way: shifting the aligned template sequence by a single residue costs nearly as much recovery as permuting it at random.

We call this *alignment fidelity*: a correspondence is faithful when each template residue sits at the query position with which it truly superposes, and its degradation is measured with every similarity signal held fixed.

We build STENCIL around this hypothesis. Rather than trusting the retrieval score, STENCIL re-aligns *every* retained candidate by exact structural superposition, keeps a *single* template, and decodes in one deterministic forward pass with 3.2M parameters and no ensembling, sampling, or test-time selection. Table 1 shows that on the split-safe CATH 4.2 test set it reaches 69.342% residue-pooled recovery, the highest among methods with runnable deterministic re-evaluations under our protocol.

Several controlled analyses isolate the mechanism and its boundary. Permuting only the aligned template tokens, with template identity, coverage, TM-score and query geometry all held fixed, re-

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.

moves 26.95 points of recovery across the full test set. A complete-case paired 2×2 analysis shows that both correspondence and selection contribute, while the direct alignment-versus-selection contrast is statistically positive and larger for alignment on the evaluated cohort. Masking every query residue to alanine before retrieval leaves the retrieved set, the alignment, and the recovery unchanged to within 0.003 points on the selected diagnostic cohort. Finally, Table 5 shows that recovery gains are strongly predicted by the sequence identity of the retrieved template, which bounds the regime in which retrieval helps.

Contributions. We formulate and test the role of residue-level alignment fidelity in retrieval-augmented inverse folding. A complete-case paired 2×2 analysis shows that both channels contribute, while the alignment-versus-selection contrast is statistically larger for alignment on the evaluated cohort. We instantiate the hypothesis as STENCIL—exhaustive exact re-alignment, one template, 3.2M parameters, and one deterministic forward pass—which attains the highest residue-pooled recovery among methods with runnable deterministic re-evaluations under our protocol. We further show that STENCIL improves substantially over direct mapped-template transfer, demonstrating that the architecture corrects template evidence rather than merely copying it (Appendix E). Finally, we document the reporting inconsistencies that make published recovery numbers non-comparable.

2 RELATED WORK

Structure-conditioned sequence design. Graph-based encoders established modern inverse folding, followed by advances in structural encoding and sequence decoding (Ingraham et al., 2019; Jing et al., 2021; Tan et al., 2022; Gao et al., 2022; Dauparas et al., 2022; Gao et al., 2023a). Subsequent work introduced pretrained protein language models and diffusion or Markov-bridge formulations (Zheng et al., 2023; Gao et al., 2024; Yi et al., 2023; Zhu et al., 2024).

Retrieval-augmented inverse folding. The closest line of work supplies retrieved structural templates or motif representations as additional conditioning. DualMPNN and Refold use explicit similarity signals to modulate template influence, while PRISM retrieves residue-level motif embeddings and aggregates them with hybrid attention (Liao et al., 2025; Zhu et al., 2026; Mahbub et al., 2026).

How STENCIL differs. These methods use similarity to determine which retrieved information reaches or influences the decoder. STENCIL instead recomputes candidate scores by exact structural superposition, uses one template, and has no learned quality gate. Section 6.2 tests the alignment and selection channels in a complete-case paired ablation and finds a statistically larger alignment contrast on the evaluated cohort.

A second difference is budget. PRISM and KW-Design combine substantially larger pretrained language or structure models with refinement, whereas STENCIL is trained from scratch with 3.2M parameters and performs one deterministic forward pass.

Evaluation of inverse folding. Inverse-folding benchmarks and aggregation conventions are often inherited rather than re-derived or stated explicitly. We revisit both issues in Section 4 and quantify their consequences in Section 5.

3 METHOD

3.1 OVERVIEW

STENCIL maps a backbone $\boldsymbol{X} \in \mathbb{R}^{L\times 4\times 3}$ with atoms N, Cα, C and O to a distribution over amino-acid sequences, using a single retrieved structural template as additional evidence. Figure 1(a) shows the pipeline: from the query backbone, structure-only retrieval, exhaustive exact re-alignment, and correspondence-indexed evidence construction, followed by the single decoding pass shown in Figure 1(c).

What the rest of the paper turns on is the residue correspondence set $\mathcal{A}$ of Eq. equation 1, drawn explicitly in Figure 1(b). Section 6 perturbs it while holding the template and every similarity score fixed.

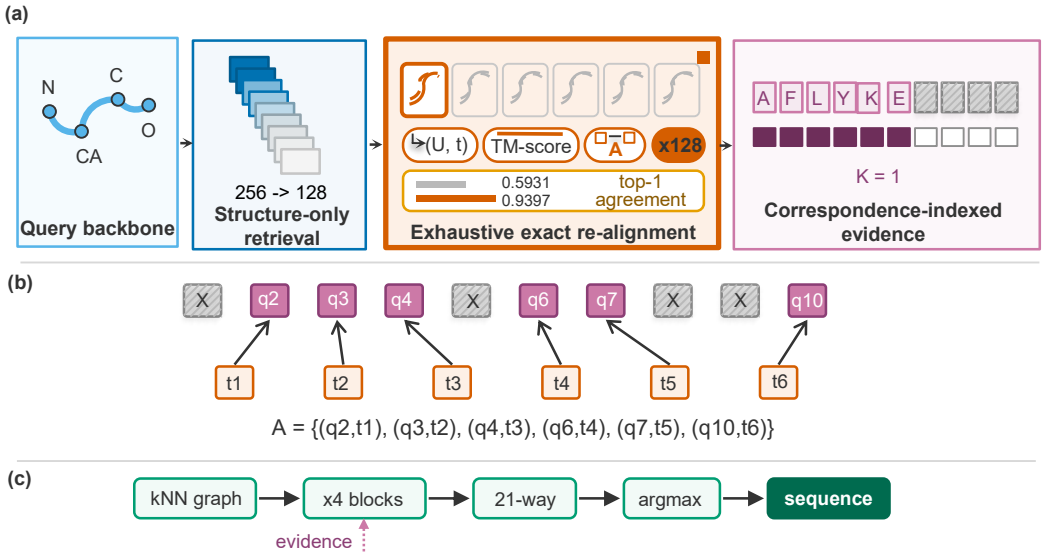

Figure 1: The STENCIL pipeline. **(a)** A query backbone $\boldsymbol{X} \in \mathbb{R}^{L\times 4\times 3}$ with atoms N, Cα, C and O is used to retrieve structural candidates with Foldseek in structure-only mode, in which no amino-acid sequence is used at retrieval. Of 256 recalled candidates, up to 128 survive the admissibility policy of Section 4, and *every* one of the retained candidates is re-aligned by exact structural superposition, yielding a rigid transform, a TM-score, and the residue correspondence $\mathcal{A}$ of Eq. 1. The single retained template then enters the model as amino-acid tokens placed at the query positions designated by $\mathcal{A}$, with a coverage mask. **(b)** The correspondence set $\mathcal{A}$, drawn as one arrow per pairing; uncovered positions carry the unknown token. **(c)** Decoding: a sparse Cα graph with four message-passing blocks, template evidence injected by cross-attention after each block, and a 21-way head decoded by deterministic arg max in one pass.

3.2 STRUCTURE-ONLY RETRIEVAL AND EXHAUSTIVE EXACT RE-ALIGNMENT

Given a query backbone, we retrieve candidate structures with Foldseek (van Kempen et al., 2024) in structure-only mode, retaining C candidates and keeping R after applying the split-safety policy of Section 4. The defining choice of STENCIL is what happens next: rather than ranking candidates by the retrieval score, we perform an *exact structural superposition* of the query against *every* retained candidate, and use the resulting alignment both to rank and to index.

Each superposition uses TM-align (Zhang & Skolnick, 2005) and yields a rigid transform $(\boldsymbol{U}, \boldsymbol{t})$, a TM-score normalised by query length, and—critically—an explicit residue correspondence set

$$\mathcal{A} \;=\; \{(q_1, t_1), \ldots, (q_m, t_m)\}, \tag{1}$$

where q_i indexes a query residue and t_i the template residue placed in correspondence with it. Candidates are then ordered by the exact-superposition TM-score, at most one candidate is kept per source structure, and the top-ranked candidate is used. STENCIL conditions on $K = 1$ template.

The pipeline cost is dominated by this stage; we report the breakdown in Section 5.2 rather than deferring it, because the hypothesis under test is precisely that this is where the budget should go.

3.3 CORRESPONDENCE-INDEXED EVIDENCE

The alignment set $\mathcal{A}$ is used to *place* template information at query positions. For each pair $(q_i, t_i) \in \mathcal{A}$ we write the template residue identity into query slot q_i; positions not covered by $\mathcal{A}$ receive a dedicated unknown token. Concretely the model receives, per query position q,

$$\text{token}[q] = \begin{cases} s_t^{\text{tmpl}} & \text{if } (q,t) \in \mathcal{A} \\ \texttt{X} & \text{otherwise} \end{cases} \tag{2}$$

$$\text{cov}[q] = \mathbf{1}\big[\exists\, t : (q,t) \in \mathcal{A}\big] \tag{3}$$

together with four scalar quality features attached to covered positions: the TM-score of the alignment, the covered fraction, a local-fit term $\exp(-d_q/5)$ in which d_q is the distance between the query Cα and the superposed template Cα, and a reciprocal-rank term.

This is the operational content of the hypothesis. The template enters the model as correspondence-indexed sequence evidence: the correspondence set $\mathcal{A}$ determines where each template residue is placed, while the decoder can use geometric context to correct or complete that evidence. The template coordinates themselves are not provided to the model. Section 6.1 perturbs $\mathcal{A}$ while holding the other evidence fields fixed.

3.4 Architecture

The backbone encoder is a sparse residue graph network: node features are backbone torsions, edge features combine radial basis expansions of inter-atomic distances with relative orientations in local frames, relative sequence position, and a chain indicator. The graph is a 32-nearest-neighbour graph over Cα atoms, symmetrised and augmented with sequence-adjacent edges, refined by four message-passing blocks of hidden width 128. Template evidence enters through a parallel stream injected after every block by cross attention, whose logits may be biased by the four quality features. The output head is a per-position projection to 21 classes. The model has 3,221,789 parameters. Training hyperparameters are in Appendix B.

3.5 Training and inference protocol

We train a single model with one seed for 30 epochs using AdamW, a cosine schedule, and token-budget batching; checkpoints are selected solely by validation recovery with validation likelihood as the tie-break. Inference is a *single* forward pass whose only inputs are the backbone and the template evidence, followed by deterministic arg max per position: no ensembling, no sampling, no temperature, no best-of-N, no test-time reranking. This protocol is held fixed for every Stencil number reported here, apart from the three-seed diagnostic of Appendix G, which is labelled as such and excluded from Table 1.

4 Experimental Setup

Data. We use the split-safe CATH 4.2 benchmark (Orengo et al., 1997; Ingraham et al., 2019; Gao et al., 2023b) and remove from training every chain whose PDB identifier appears in the validation or test split, even when its sequence differs; in addition, when a native sequence appears in more than one split we retain a single copy, dropping the duplicate from the test split so that the validation cohort used for checkpoint selection is untouched. Post-filter counts and the full audit are in Appendix A. Templates are retrieved from a structure database whose local snapshot is dated 2026-07-10.

Leakage control. Template admissibility is enforced by an explicit, auditable policy rather than by relying on the transitivity of the benchmark split. A candidate is rejected if its structure identifier or its exact sequence hash appears in the validation/test blocklist, if it shares an identifier or exact sequence with the query, if its TM-score to the query is at least 0.99, or if another candidate from the same source structure has already been retained. The packed evidence used for all reported results contains zero blocked, zero same-query, and zero near-copy references.

Metrics. Let protein p have L_p scored residues of which C_p are recovered. We report three aggregations, because the field uses them interchangeably without stating which:

$$\text{micro} = \frac{\sum_p C_p}{\sum_p L_p}, \qquad \text{macro} = \frac{1}{P}\sum_p \frac{C_p}{L_p}, \qquad \text{median} = \text{median}_p \, \frac{C_p}{L_p}. \tag{4}$$

These are not interchangeable and cannot be converted into one another. We designate **micro** as the primary metric: its weighting is fully determined by the scoring code and can be verified from released artifacts, whereas the median reported in several papers could only be identified by re-deriving it from per-protein outputs. Negative log-likelihood uses the same residue-pooled denominator as micro recovery. Every likelihood is conditioned only on inference-time information—the backbone, retrieved templates, or a retrieved alignment—and never on the native sequence. Because these

conditioning sets differ across methods, the resulting likelihoods are not strictly identical; we nevertheless follow the literature in comparing the confidence assigned to the correct sequence under each method's permitted inputs.

Unified re-evaluation of baselines. Published recovery numbers for this benchmark are not directly comparable: the scored-residue denominator, the decoding protocol, and the aggregation all vary, and are often unstated. We therefore re-evaluate every baseline for which artifacts permit it, using the same benchmark cohort and coordinate-validity mask, and report all three aggregations. Methods are grouped by what the original release makes possible. Released checkpoints are evaluated directly. Methods with training code but no benchmark checkpoint are retrained on this split. Methods whose only checkpoint was trained on a different corpus are excluded, because such a checkpoint may overlap the test cohort and cannot be scored on this split without contaminating it. Methods that release neither code nor checkpoints are quoted from their papers and marked as such.

Every re-evaluated baseline is decoded exactly once, deterministically, with no sampling and no selection over samples imposed by us; any iteration a method performs inside its own forward pass is left exactly as published.

5 Results

5.1 Recovery under a unified evaluation protocol

Under residue-pooled recovery Stencil is 3.6 points above the strongest runnable baseline under our deterministic protocol. The same ordering holds under per-protein averaging. Under the median the ordering changes and KW-Design is ahead by 1.1 points. We report all three aggregations rather than selecting the one that favours us; Section 6.5 shows that the divergence tracks the retrieval-augmented rows rather than noise.

Table 1 also shows why published numbers are not directly comparable. Under identical data, code, and evaluation, our two retrained ProteinMPNN recipes differ by 7.4 points. Among released-weight reruns with matched median aggregation, our PiFold and ProteinMPNN-CMLM results reproduce the printed values within 0.002 and 0.76 points, yet published ProteinMPNN tables span 40.27–49.87. This discrepancy motivates the unified protocol; detailed provenance is in Appendix C.

Table 1: Native sequence recovery on the CATH 4.2 test set. *Reported* columns quote the originating paper; *re-evaluated* columns are measured by us under one protocol. *RAG* marks structural template retrieval, so a method that samples a sequence alignment but retrieves no structure is marked no. *Weights* records where the evaluated parameters came from. Method sources and row-level provenance are provided in Appendix C. Best and second best per re-evaluated column in bold.

			Recovery (%)				PPL	
Method	RAG	Weights	reported	micro	macro	median	reported	re-eval.
ProteinMPNN	no	retrained	–	40.835	38.888	40.779	–	6.671
ProtMPNN-CMLM	no	released	48.62	47.659	45.550	47.861	5.03	5.192
ProteinMPNN-PIB	no	retrained	45.96	48.209	46.116	48.070	4.61	5.174
PiFold	no	released	51.66	51.603	49.392	51.658	4.55	4.556
LM-Design	no	retrained	54.41	54.926	51.876	54.515	4.41	4.519
KW-Design	no	released	60.77	58.972	56.049	**59.442**	3.46	3.600
DualMPNN	yes	retrained	65.51	**65.700**	**63.117**	53.535	3.19	**3.217**
Refold	yes	none	63.00	–	–	–	3.48	–
PRISM	yes	none	60.43	–	–	–	2.71	–
Stencil	yes	–	–	**69.342**	**65.748**	**58.350**	–	**2.844**

Cohort sizes and per-method caveats: Appendix C.

5.2 End-to-end cost

Exhaustive exact re-alignment dominates the pipeline. On 100 length-stratified test queries, using a single T4 GPU with 10 CPU cores and taking the median of three runs, the re-alignment stage alone accounts for 71.6% of warm-cache end-to-end time, and the full pipeline is 5.4× costlier than taking

the retrieval alignment directly. This corresponds to roughly 3.0 seconds per query with a warm structure cache and 6.4 seconds cold, against 0.11 seconds for the cached-evidence forward pass alone. The stage-by-stage breakdown is in Appendix B. Section 6.2 quantifies what that expenditure buys.

5.3 Cross-dataset generalisation on TS50 and TS500

TS50 and TS500 (Li et al., 2014) are reported transfer results collected from the originating papers; we did not re-evaluate these baselines under our unified protocol. Because the benchmarks contain training overlap and aggregation conventions are not uniformly specified, we treat the reported values as secondary reference context rather than as a like-for-like comparison. The Stencil row reports the canonical single-checkpoint micro-AAR and micro-PPL under the protocol used throughout; it is not ranked or boldfaced against the reported rows.

TS benchmarks contain training overlap. We report them only as secondary transfer context; the overlap and identity-exclusion audit is deferred to Appendix C.

The Stencil TS results use the same single-checkpoint model and alignment configuration as the main CATH baseline; only the evaluation benchmark changes.

Table 2: Reported and reference TS50/TS500 recovery and PPL. Baseline rows are from the originating papers; aggregation and protocols are not uniform, so the rows are reference context rather than a ranking. STENCIL reports canonical single-checkpoint micro values.

	TS50		TS500	
Method	AAR	PPL	AAR	PPL
ProteinMPNN	54.43	3.93	58.08	3.53
ProtMPNN-CMLM	53.68	3.46	56.45	3.35
PiFold	58.72	3.86	60.42	3.44
LM-Design	56.92	3.82	64.50	2.13
Refold	60.00	3.78	63.00	3.51
PRISM	67.92	2.43	70.53	2.27
DualMPNN	70.99	2.76	70.37	2.71
Stencil	81.55	1.90	77.57	2.10

6 Analysis

This section reports post-hoc diagnostics under two contexts: fixed-checkpoint validation interventions in Sections 6.1–6.2, and frozen CATH 4.2 test diagnostics in Sections 6.3–6.6. Full-test estimates are identified where used; no test result selected the checkpoint, threshold, budget, or protocol.

6.1 Alignment fidelity is necessary

Table 3: Mechanistic ablations of correspondence-indexed template evidence. Copy rows replace covered template residues and differ only in the prediction used at uncovered positions. Shift and global shuffle corrupt correspondence while retaining the template source.

Condition	micro	macro	median	Δ micro
Stencil / full evidence	73.07	70.37	69.07	0.00
Copy covered / Stencil uncovered	66.64	63.64	67.87	-6.43
Copy covered / geometric fallback uncovered	66.59	63.60	67.54	-6.48
Empty evidence	42.89	41.48	43.17	-30.18
One-residue shift	41.51	40.15	41.25	-31.56
Composition-preserving global shuffle	40.54	39.42	40.68	-32.53

Table 3 separates the roles of template evidence and its registration. Full evidence reaches 73.07%, whereas removing the template reduces recovery to 42.89%, showing that the retrieved evidence is itself informative (cf. Appendix D). Directly copying template residues at covered positions reaches 66.59% with the geometric fallback and 66.64% when the full Stencil prediction is retained at uncovered positions. The 0.05-point difference between these two copy-covered controls is small, whereas restoring the full model adds 6.43 points. In this control, the learned gain is therefore concentrated on correcting covered template evidence. The one-residue shift and composition-preserving global shuffle reduce recovery to 41.51% and 40.54%, respectively; Figure 2(a) extends these endpoint controls into a continuous corruption curve.

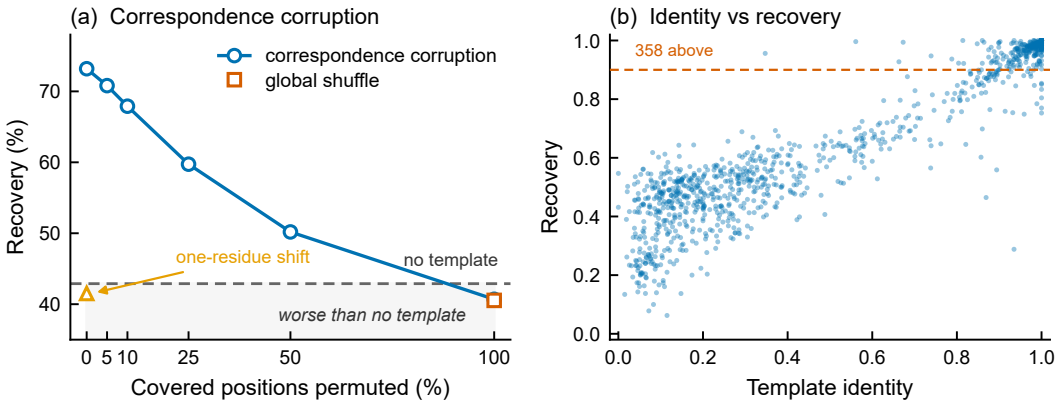

Figure 2: **(a)** Alignment fidelity is necessary. Dose-response of residue-pooled recovery as the fraction of covered template tokens permuted increases. The dashed line and open markers are the synchronized no-template, global-shuffle, and one-residue-shift references described in Section 6.1; the dose-response curve is a separate validation export. **(b)** Per-protein recovery against the sequence identity of the retrieved top-1 template. The high-recovery population lies almost entirely above 0.9 identity; four queries without a retained template are omitted.

We test correspondence fidelity by permuting the template tokens placed at covered query positions. Figure 2(a) shows the resulting experiment. Its dose-response curve uses a separate validation export in which each query's admissible top-1 template is held fixed across conditions; only the token-to-position mapping changes. The no-template, one-residue-shift, and global-shuffle markers use the synchronized fixed-checkpoint intervention rerun reported in Table 3; they are reference points, not additional dose conditions.

The dose-response curve falls from 73.19% with exact correspondence to 40.69% under full permutation, a 32.50-point drop. This is the separate validation dose-response export, not the primary full-test estimate: a full-test shuffle analysis gives a 26.95-point drop. A paired bootstrap over the query-level clean-versus-permuted differences gives a 95% confidence interval of $[25.25, 28.55]$ points. Within the dose-response curve, corruption is graded rather than threshold-like: corrupting 5% of covered positions already costs 2.38 points, and the curve declines monotonically.

In the synchronized intervention rerun, the no-template reference is 42.8932%, the composition-preserving global-shuffle point is 40.5363%, and the one-residue shift yields 41.5080%. A one-notch misregistration is therefore more destructive than discarding the correspondence entirely.

6.2 Alignment source versus template selection

Exact re-alignment could help for two distinct reasons: it may select a better template, or it may place the selected template more accurately. We separate these with the 2×2 ablation in Table 4, holding the checkpoint fixed. The complete-case paired cohort includes only queries for which all four cells have a selected template and valid mapping evidence. Rows use either the retrieval score or the exact-superposition score to select the top-1 template; columns supply either the Foldseek or exact TMalign correspondence to the model.

Table 4: Complete-case paired validation ablation with a frozen checkpoint. Rows vary the top-1 selection score; columns vary the residue mapping supplied to the model.

Select by	Foldseek map	Exact map
Retrieval score	70.726	74.823
Exact score	72.809	**75.386**

On this complete-case cohort, replacing exact correspondence with retrieval correspondence at exact selection lowers micro recovery by 2.58 points, whereas replacing exact selection with retrieval-score selection at exact correspondence lowers it by 0.56 points. Their paired contrast is $+2.01$ points with a 95% query-level paired-bootstrap interval of $[0.98, 3.14]$ and $p = 0.0001$. We therefore conclude

that the alignment contrast is statistically larger on this cohort, while retaining the narrower claim that template selection remains an independent channel.

We do not interpret this contrast as a universal effect ratio. It is an estimand for queries with complete evidence under both selection sources; cases in which the retrieval branch has no selected template are handled separately as template-availability failures rather than as pure selection-source comparisons.

6.3 Retrieval does not exploit query sequence identity

A natural objection is that the pipeline locates near-homologues using query residue identities. We test this on a post-hoc cohort by replacing every query residue with alanine before retrieval and rerunning the unchanged pipeline.

On the 358 highest-recovery queries the retained evidence and recovery are unchanged to within 0.003 points, evidence against a query-side shortcut. This is a query-side diagnostic, not a full audit of retrieved-template homology.

6.4 Where retrieval helps, and where it does not

Whether a target enters the high-recovery regime is strongly predicted, in this post-hoc analysis, by the sequence identity of the retrieved template, with an AUC of 0.985. Table 5 reports the corresponding univariate comparisons, and Figure 2(b) shows the same separation without any fitted threshold. This identity is an analysis-time diagnostic, not an inference-time routing signal. The result bounds the claim of Section 6.1: alignment fidelity is necessary, but it operates on templates that must first be available. When no high-identity template exists, conditioning on the best available low-identity template is not merely unhelpful. On the paired low-identity subset, where a non-empty top-1 template has sequence identity below 20%, it underperforms a purely geometric model trained without templates, 42.75% against 45.88%.

Table 5: Univariate prediction of high recovery (> 0.9) from top-1 template properties. High and Other give mean property values in the two outcome groups; AUC is ROC-AUC with a 95% bootstrap interval. Sequence identity is a post-hoc diagnostic.

Property	High	Other	AUC [95%]
Sequence identity	0.9688	0.3148	**0.985** [0.979,0.991]
TM-score	0.9569	0.7767	0.908 [0.890,0.925]

This analysis identifies two conditions for retrieval gains: the database must contain a high-identity structural template, and the method must register that template faithfully. The first is a property of the database and target; the second is what a method can control. The identity measurement is diagnostic only, whereas Stencil is designed around the second condition.

The actionable condition is registration: once an informative template exists, recovery responds more to how faithfully the template's residues are mapped onto the target than to how the template was selected; both channels contribute.

6.5 Micro–median divergence under retrieval

Table 6: Separation between residue-pooled and median recovery. The two retrieval-augmented methods show an order-of-magnitude larger separation than non-retrieval methods, independently of capacity or absolute performance. The two further re-evaluated baselines omitted from the table for space also fall within half a point of zero: -0.20 for ProtMPNN-CMLM and $+0.06$ for the official-recipe ProteinMPNN retraining.

Method	Retrieval	micro	median	micro − median
Stencil	yes	69.342	58.350	**+10.992**
DualMPNN	yes	65.700	53.535	**+12.165**
KW-Design	no	58.972	59.442	-0.47
LM-Design	no	54.926	54.515	$+0.41$
PiFold	no	51.603	51.658	-0.06
ProteinMPNN (PIB recipe)	no	48.209	48.070	$+0.14$

Table 6 shows that both retrieval-augmented methods have a separation of roughly 11–12 points; all non-retrieval rows fall within 0.5 points of zero, including the largest model in the table. The effect therefore does not track capacity.

The cause is distributional, as Figure 3 shows. Retrieval produces a bimodal recovery distribution: a high-recovery subgroup lifts the residue-pooled mean above the median. The dispersion account, quantified for the micro–macro component in Appendix G, attributes the gap to performance spread rather than length weighting.

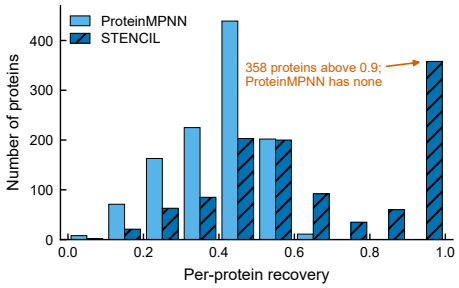

Figure 3: Per-protein recovery distributions. STENCIL has 358 test proteins above 0.9 recovery; ProteinMPNN has none above 0.7.

6.6 WHY WE DO NOT ENSEMBLE

The heterogeneity account predicts that variance reduction should improve the median more than the mean. A three-seed probability ensemble does so, consistent with this mechanistic diagnostic, but is excluded from Table 1 because it triples the training budget and is not comparable to single-model entries.

7 DISCUSSION AND LIMITATIONS

Scope of the claim. Our claim is comparative and bounded: on a complete-case paired validation cohort, the alignment contrast is statistically larger than the selection contrast. We do not claim that template similarity is irrelevant—template selection remains an independent channel—nor that retrieval generalises uniformly to targets without informative templates. The identity analysis and stress tests instead define the regime in which the retrieved evidence is useful.

Components that do not earn their place. The quality features are dispensable at $K = 1$, and additional template rows are largely redundant under the present evidence construction. This favours efficiency but leaves the method underdeveloped for remote-homology settings that require complementary multi-template evidence; detailed values are in Appendix F.

Evaluation scope. The main comparison is on CATH 4.2; TS50 and TS500 are secondary transfer context. Sections 6.1–6.2 report fixed-checkpoint validation interventions, whereas Sections 6.3–6.6 report frozen-test post-hoc diagnostics. Test results did not determine model or protocol choices.

One metric where we are not first. Under median per-protein recovery, KW-Design is ahead of STENCIL. We report all three aggregations rather than the one that favours us. We regard the residue-pooled figure as the primary metric for the reasons given in Section 4.

Residual homology risk. Our exact-match blocklist does not remove all near-homologous retrieved structures with different identifiers or non-identical sequences. The post-hoc identity analysis of Table 5 and Figure 2(b) therefore bounds, but does not eliminate, this residual risk.

Outlook. A future inference-time proxy could route to a template-free model when retrieved evidence is uninformative. Extending the template pool to predicted structures could enlarge the applicable regime but would require additional contamination controls.

8 CONCLUSION

Retrieval-augmented inverse folding is usually framed as a retrieval problem: find more templates and weight them by similarity. Template retrieval makes informative structural evidence available; precise residue registration determines how much of it becomes sequence recovery. A complete-case paired analysis finds a statistically larger alignment contrast than selection, while permuting only aligned tokens costs 26.95 recovery points. Building around this yields a 3.2M-parameter, single-template model with the highest residue-pooled recovery among methods with runnable deterministic re-evaluations under our protocol. The identity analysis defines the boundary: retrieval is useful when informative template evidence is available.

AI USE STATEMENT

The authors used generative AI assistance for writing support and language polishing, and for retrieval and discovery of related work. AI assistance was used to improve clarity and organization of the manuscript and to help identify relevant literature; it did not replace the authors' verification of scientific claims, numerical results, or citations. The authors did not use generative AI to generate experimental data or substitute for the reported experiments. All AI-assisted text and literature suggestions were reviewed by the authors, who remain fully responsible for the manuscript, its claims, its numerical results, and its supplementary artifacts.

ETHICS STATEMENT

This work designs amino-acid sequences for given protein backbones and is evaluated entirely on public structural databases. No human subjects, personally identifiable information, or newly collected data are involved. Sequence recovery against native proteins is an in-silico proxy and does not establish that designed sequences fold, function, or are safe to express; we make no claim of experimental validity. As with any protein design method, improved sequence design capability is dual-use. The models and analyses reported here operate on single-chain backbones from an established public benchmark and do not target any specific biological function.

REPRODUCIBILITY STATEMENT

Section 4 specifies the benchmark, the scored-residue denominator, and the three aggregations used throughout. Section 3.5 specifies the training and inference protocol, including the single-seed, single-pass, no-ensembling discipline applied to every reported STENCIL number apart from the labelled three-seed diagnostic of Appendix G. The template-admissibility policy is enumerated in Section 4 and its audit counts are given in Appendix A. Retrieval and alignment hyper-parameters and the structure-database snapshot date are listed in Appendix B. The unified re-evaluation protocol applied to baselines, including which methods were evaluated from released checkpoints, which were retrained, and which could only be quoted, is described in Section 4 and itemised in Appendix C.

An anonymized repository containing the code, configuration files, processed inputs, and scripts needed to reproduce the reported results will be provided at `https://anonymous.4open.science/r/stencil`.

REFERENCES

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, et al. Robust deep learning-based protein sequence design using ProteinMPNN. *Science*, 378(6615): 49–56, 2022. doi: 10.1126/science.add2187.

Zhangyang Gao, Cheng Tan, and Stan Z. Li. Alphadesign: A graph protein design method and benchmark on AlphaFoldDB. *arXiv preprint arXiv:2202.01079*, 2022.

Zhangyang Gao, Cheng Tan, and Stan Z. Li. PiFold: Toward effective and efficient protein inverse folding. In *International Conference on Learning Representations*, 2023a. URL `https://openreview.net/forum?id=oMsN9TYwJ0j`.

Zhangyang Gao, Cheng Tan, Yijie Zhang, Xingran Chen, Lirong Wu, and Stan Z. Li. Proteininvbench: Benchmarking protein inverse folding on diverse tasks, models, and metrics. In *Advances in Neural Information Processing Systems*, 2023b. URL `https://proceedings.neurips.cc/paper_files/paper/2023/hash/d73078d49799693792fb0f3f32c57fc8-Abstract-Datasets_and_Benchmarks.html`.

Zhangyang Gao, Cheng Tan, Xingran Chen, Yijie Zhang, Jun Xia, Siyuan Li, and Stan Z. Li. KW-Design: Pushing the limit of protein design via knowledge refinement. In *International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=mpqMVWgqjn`.

John Ingraham, Vikas K. Garg, Regina Barzilay, and Tommi S. Jaakkola. Generative models for graph-based protein design. In *Advances in Neural Information Processing Systems*, pp. 15794–15805, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/f3a4ff4839c56a5f460c88cce3666a2b-Abstract.html.

Bowen Jing, Stephan Eismann, Patricia Suriana, Raphael J. L. Townshend, and Ron Dror. Learning from protein structure with geometric vector perceptrons. In *International Conference on Learning Representations*, 2021.

Zhixiu Li, Yuedong Yang, Eshel Faraggi, Jian Zhan, and Yaoqi Zhou. Direct prediction of profiles of sequences compatible with a protein structure by neural networks with fragment-based local and energy-based nonlocal profiles. *Proteins: Structure, Function, and Bioinformatics*, 82(10): 2565–2573, 2014. doi: 10.1002/prot.24620.

Xuhui Liao, Qiyu Wang, Zhiqiang Liang, Liwei Xiao, and Junjie Chen. DualMPNN: Harnessing structural alignments for high-recovery inverse protein folding. In *Advances in Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=R42O6v84cX.

Sazan Mahbub, Souvik Kundu, and Eric P. Xing. PRISM: Enhancing protein inverse folding through fine-grained retrieval on structure-sequence multimodal representations. In *International Conference on Learning Representations*, 2026.

C. A. Orengo, A. D. Michie, S. Jones, D. T. Jones, M. B. Swindells, and J. M. Thornton. CATH – a hierarchic classification of protein domain structures. *Structure*, 5(8):1093–1109, 1997. doi: 10.1016/s0969-2126(97)00260-8.

Cheng Tan, Zhangyang Gao, Jun Xia, Bozhen Hu, and Stan Z. Li. Generative de novo protein design with global context. *arXiv preprint arXiv:2204.10673*, 2022.

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. *Nature Biotechnology*, 42(2):243–246, 2024. doi: 10.1038/s41587-023-01773-0.

Junde Xu, Zijun Gao, Xinyi Zhou, Jie Hu, Xingyi Cheng, Le Song, Guangyong Chen, Pheng-Ann Heng, and Jiezhong Qiu. Protein inverse folding from structure feedback. *arXiv preprint arXiv:2506.03028*, 2025.

Kai Yi, Bingxin Zhou, Yiqing Shen, Pietro Liò, and Yu Guang Wang. Graph denoising diffusion for inverse protein folding. In *Advances in Neural Information Processing Systems*, 2023.

Yang Zhang and Jeffrey Skolnick. TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Research*, 33(7):2302–2309, 2005. doi: 10.1093/nar/gki524.

Zaixiang Zheng, Yifan Deng, Dongyu Xue, Yi Zhou, Fei Ye, and Quanquan Gu. Structure-informed language models are protein designers. In *International Conference on Machine Learning*, 2023.

Yiheng Zhu, Jialu Wu, Qiuyi Li, Jiahuan Yan, Mingze Yin, Wei Wu, Mingyang Li, Jieping Ye, Zheng Wang, and Jian Wu. Bridge-IF: Learning inverse protein folding with markov bridges. *arXiv preprint arXiv:2411.02120*, 2024.

Yiran Zhu, Changxi Chi, Hongxin Xiang, Wenjie Du, Xiaoqi Wang, and Jun Xia. Refold: Refining protein inverse folding with efficient structural matching and fusion. *arXiv preprint arXiv:2603.14350*, 2026.

A Data and leakage audit

The native CATH 4.2 single-chain split contains the training, validation, and test cohorts used in this paper. To construct the split-safe training pack, we removed from the training split every chain whose PDB identifier appears in the validation or test split, even when the sequences differ. This first filter reduces the packed training split; the validation split is unchanged. The reported Stencil test result uses the coordinate-valid, audit-defined evaluated cohort. Baseline re-evaluation uses its own coordinate-validity cohort, whose exact denominator is recorded in Appendix C.

Table 7: Packed-evidence audit for the CATH 4.2 splits used in this paper. The four violation categories are the ones enforced by the admissibility policy of Section 4; all are zero by construction and verified after packing.

Split	Queries	Template rows	Empty	Blocked	Same-query	Near-copy	Non-finite
train	17,918	558,008	85	0	0	0	0
validation	608	18,966	2	0	0	0	0
test	1,119	34,920	4	0	0	0	0

The six admissibility predicates applied to every retrieved candidate are: rejection if the candidate structure identifier appears in the validation or test blocklist; rejection if its exact sequence hash appears in that blocklist; rejection if it shares a structure identifier with the query; rejection if it shares an exact sequence with the query; rejection if its TM-score normalised by query length is at least 0.99; and rejection if a candidate from the same source structure has already been retained.

The standard CATH 4.2 test split contains 1,120 chains. Exactly one of them, `3sk2.A`, shares a complete native sequence with a validation chain and is therefore removed by the split-safety policy of Section 4, which drops every chain whose sequence overlaps another split rather than only those sharing a PDB identifier. The remaining 1,119-query cohort, $181{,}528$ scored residues in total, is the one on which every reported STENCIL test figure is computed; the four queries marked empty in Table 7 are the separate question of having no admissible template, not of being excluded from the split.

B RETRIEVAL, ALIGNMENT, AND EVALUATION PROTOCOL

Retrieval. Structure-only search mode; 256 candidates requested per query; up to 128 retained after the admissibility policy; e-value threshold disabled; output fields restricted to query, target, alignment TM-score, sequence identity, and alignment length. The structure database snapshot is dated 2026-07-10 and indexes 1,028,002 valid protein chains. That snapshot is built from 241,467 local `.ent.gz` structure files; parsing those at chain granularity yields 1,951,296 candidate chains, of which 920,975 are discarded as too short and 2,319 as non-protein, leaving the 1,028,002 chains that are actually indexed.

Exact re-alignment. Every retained candidate is superposed against the query. Candidates are ordered by TM-score normalised by query length, ties broken deterministically by canonical identifier; at most one candidate is kept per source structure; the top 32 are packed and the top 1 is used at inference.

Training. AdamW, cosine schedule with minimum learning-rate ratio 0.1, initial learning rate 5×10^{-4}, weight decay 10^{-4}, token-budget batching at 700 tokens, 30 epochs, mixed precision, single seed. Checkpoints are selected by validation recovery with validation likelihood as the tie-break; test data is never accessed during selection.

Table 8: End-to-end cost decomposition over 100 length-stratified test queries on a single T4 GPU with 10 CPU cores, median of three runs. Totals are for the full 100-query batch. The Foldseek-only row is a cost reference; its recovery is not reported because the controlled comparison is Table 4.

Configuration	Cold cache	Warm cache	Share of warm total
Full pipeline	637.1 s	301.5 s	100 %
exact re-alignment stage	–	216.0 s	71.6 %
Foldseek-only alignment	390.1 s	56.2 s	–
Cached-evidence forward pass only	–	10.7 s	3.6 %

C BASELINE RE-EVALUATION DETAILS

Why the CMLM row is retained. The decoding column records a conditional masked LM for ProteinMPNN-CMLM, and that should not be read as conditioning on the native sequence. Its like-

Table 9: Provenance of each re-evaluated baseline. Tier A uses the released checkpoint unmodified. Tier B retrains on the CATH 4.2 split because no benchmark checkpoint is released. We report no baseline whose only available checkpoint was trained on a different corpus: such a checkpoint may overlap the test cohort, which is favourable to that baseline, and we cannot bound the size of that advantage. One ProteinMPNN checkpoint of this kind was evaluated early on and is deliberately excluded from Table 1 for this reason.

Method	Tier	Weights	Decoding
PiFold	A	official checkpoint	one-shot argmax
KW-Design	A	official checkpoint	3 recycles, confidence select
ProteinMPNN-CMLM	A	official checkpoint	conditional masked LM
LM-Design, frozen	A/B[†]	official, adapter retrained	iterative refinement
ProteinMPNN, PIB recipe	B	retrained on CATH 4.2	unconditional and AR
ProteinMPNN, official recipe	B	retrained on CATH 4.2	unconditional and AR
DualMPNN	B	retrained on CATH 4.2	AR

[†] No checkpoint is released for this variant. Both encoders are the official frozen checkpoints and only the structural adapter is retrained, so the row is a controlled reconstruction rather than a checkpoint reload. Our figures land 0.52 points of residue-pooled recovery and 0.11 in likelihood from the ones the originating paper prints for it.

lihood is evaluated under the same backbone-only, full-mask conditioning as the other rows; "conditional" refers to the model conditioning on its own partially predicted sequence during iterative decoding, which every row of this table does in some form. The comparator excluded on these grounds is different: its forward pass consumes the native sequence directly, so the figure we hold for it is teacher-forced and not comparable. The exclusion criterion is therefore applied consistently.

Two ProteinMPNN recipes. Retraining the same architecture on the same data with the same evaluation, changing only the training recipe, moves recovery by 7.4 points. The official configuration uses 48 neighbours, backbone noise 0.2, a Noam schedule, and 150 epochs for roughly 71k optimiser steps. The third-party configuration adopted by the benchmark literature uses 30 neighbours, no backbone noise, a constant learning rate of 10^{-3}, and 100 epochs for roughly 225k steps. The figure usually quoted for ProteinMPNN on this benchmark corresponds to the latter.

An unreproducible released result. KW-Design releases both a checkpoint and the code that produced its published numbers, so it can be re-run rather than quoted. Re-running them gives 59.442 median recovery against the 60.77 the paper reports for the same split, a gap of 1.33 points. The same checkpoint's own logged validation likelihood is 1.2909, while we obtain 1.32277 with batch size one and 1.32175 through the released validation routine. The validation offset of $+0.031$ matches the test offset of $+0.0316$, both taken against the likelihood logged with that checkpoint rather than against the perplexity printed in the paper, and is independent of batch size, which places the discrepancy in the trainable module rather than in our evaluation: the three frozen components were verified to produce bitwise identical outputs.

Identifying an unstated aggregation. One widely cited baseline reports a single recovery figure per benchmark without naming the aggregation. Its three published figures match our reproduction's per-protein median to within 0.05 points on all three benchmarks, and do not match its residue-pooled value. We therefore treat that paper's figures as medians.

Comparators deliberately omitted. Three methods that appear in most tables of this literature are absent from Table 1, and we state why rather than leaving the omission silent. For one of them the released inference path is multinomial sampling at temperature 0.1, so the only figure we can produce for it is a sampling figure; including it would break the deterministic decoding that every other row obeys. The teacher-forced argmax figure we do hold for it is diagnostic only, because its forward pass consumes the native sequence, so it is not an autoregressive greedy decode and is not comparable to ours. A second releases a checkpoint for a different CATH version and none for CATH 4.2, so it cannot be scored on our cohort without substituting a different test set. The third releases weights and code but we did not re-evaluate it, and we prefer to quote a number we did not measure only where the method is needed for the argument; it is named in Section 2 instead.

C.1 Cross-paper consistency of published baseline figures

Baseline rows in this literature are largely copied forward from earlier tables rather than recomputed. Table 10 collects the published CATH 4.2 figures as printed in the tables of Xu et al. (2025), Mahbub et al. (2026), Liao et al. (2025) and Zhu et al. (2024); these four are the papers whose printed tables supply the quoted cells, and the citations record that provenance rather than any claim about their own methods. Most rows correspond to comparator configurations used in Table 1; the GraDe-IF rows are kept because they show a different failure mode: the two sources agree exactly wherever both print a number, while one prints the perplexity columns only and leaves all three recovery columns blank, so a method can be carried between tables with its recovery figures simply missing. Some methods are quoted identically everywhere; others are not. The LM-Design entry is restricted to the pretrained ProtMPNN-CMLM: freeze variant reproduced in this paper.

Table 10: CATH 4.2 figures as printed in four baseline tables. Short and single-chain columns are included because they localise where the disagreements occur. A dash means the row is absent from that table or that we could not read that cell reliably from the published version.

		PPL			Recovery (%)		
Method	Printed in	short	s.c.	all	short	s.c.	all
Consistent across sources							
PiFold	Xu et al. (2025)	6.04	6.31	4.55	39.84	38.53	51.66
	Mahbub et al. (2026)	6.04	6.31	4.55	39.84	38.53	51.66
	Liao et al. (2025)	6.04	6.31	4.55	39.84	38.53	51.66
GraDe-IF	Liao et al. (2025)	5.49	6.21	4.35	45.27	42.77	52.21
	Zhu et al. (2024)	5.49	6.21	4.35	–	–	–
Inconsistent across sources							
ProteinMPNN	Xu et al. (2025)	8.82	9.17	5.41	28.83	27.20	40.27
	Mahbub et al. (2026)	6.21	6.68	4.61	36.35	34.43	45.96
	Liao et al. (2025)	6.21	6.68	4.57	36.35	34.43	49.87
Single source							
LM-Design	Mahbub et al. (2026)	7.01	6.58	4.41	35.19	40.00	54.41

Two observations follow. First, the recovery figure printed for ProteinMPNN on this split ranges from 40.27 to 49.87, a spread of 9.60 points, which is larger than the margin most papers in this literature claim over their nearest competitor. Second, the cellwise pattern identifies the ProteinMPNN inconsistency. The lines from Mahbub et al. (2026) and Liao et al. (2025) are numerically identical in four of six cells and differ only in the two all-chain cells, by 0.04 in likelihood and 3.91 points in recovery. A row recomputed independently would be expected to move in every cell; agreement in four cells and disagreement in two is the signature of a row carried over with two entries replaced. For LM-Design, this paper retains only the pretrained ProtMPNN-CMLM: freeze configuration reproduced in the main table, so the cross-paper audit reports its 54.41 reference value only.

We draw no conclusion about which figure is correct. The consequence for this paper is procedural: a quoted baseline number carries no protocol, so we quote the figure belonging to the specific variant we reproduced, we re-evaluate every method for which weights or predictions are available, and we report all three aggregations rather than one.

A related case is internal to a single paper. One comparator's main table prints an all-chain likelihood of 3.18 while its own text and abstract state 3.19. We quote 3.19 in Table 1 because that is the figure the authors state in prose, and we note the discrepancy here rather than silently selecting one.

Notes on Table 1. Every re-evaluated column uses the coordinate-validity mask of Section 4; exact denominators are reported with the corresponding evaluation artifacts. The *Weights* column records where the evaluated parameters came from. *released* means the official checkpoint was loaded and evaluated as published. *retrained* means no checkpoint for this split is released, so we trained the method on CATH 4.2 from its released code. This includes LM-Design, for which no checkpoint is released for the variant we reproduce at all: both encoders are its official frozen checkpoints and only the structural adapter is retrained by us, so that row is a controlled reconstruction rather than a checkpoint reload, and it lands 0.52 points of residue-pooled recovery and 0.11 in likelihood from

the figures its own paper prints. *none* means neither weights nor code are released, so we quote the paper and re-evaluate nothing.

One caveat attaches to a single row. The strongest non-retrieval baseline is marked no under RAG even though it samples a multiple sequence alignment at inference: what it retrieves is sequence, not structure, and no structure is aligned to the query.

Table 11: Canonical TS micro-AAR under progressively stricter identity exclusion. All rows use the single H128/L4 checkpoint of Section 3.4 (hidden width 128, 4 layers) at $K = 1$, trained with the single seed used for every STENCIL number in this paper, and the same structure-only retrieval and exact re-alignment protocol as the main results; only the evaluated cohort changes. These values are secondary stress test context and are not a like-for-like ranking against the reported literature rows in Table 2.

	TS50		TS500	
Cohort	n	micro AAR	n	micro AAR
Unfiltered	50	81.55	500	77.57
Exact-overlap removed	28	84.42	470	77.47
Similarity to train $< 40\%$	14	81.06	174	75.75
Similarity to train $< 30\%$	14	81.06	152	75.32

Two features of this table are worth stating explicitly rather than leaving to the reader. First, removing exact-overlap queries *raises* the TS50 figure, from 81.55 on 50 queries to 84.42 on 28, while lowering TS500 by 0.10; overlap is therefore not what inflates the TS numbers, and the exclusion is reported as a sensitivity check rather than as a correction. Second, the $< 40\%$ and $< 30\%$ strata coincide on TS50 (both $n = 14$ at 81.06) because no TS50 target has a similarity to the training set inside the 30–40% band, so the two thresholds select the same queries there; on TS500 they differ (174 versus 152).

Train-similarity sensitivity. The strata above are defined by the sequence similarity between each TS query and the CATH 4.2 training set, not by any property of the retrieved template. A stricter threshold keeps only those TS queries that are increasingly dissimilar to anything the model was trained on, so these rows are a training-set contamination audit rather than a statement about template availability. They are reported as secondary context because the TS benchmarks do carry such overlap—of the 500 TS500 queries, 30 are exact overlaps with a training chain and, of the remaining 470, a further 296 have at least 40% sequence similarity to one—and are therefore not a clean out-of-distribution measurement.

D ADDITIONAL INTERVENTIONS

Template-free reference used in Section 6.4. The purely geometric model trained without templates attains 44.511% residue-pooled recovery on the full CATH 4.2 evaluation cohort, against 69.342% for STENCIL. The 45.88% quoted in Section 6.4 is its score on the low-identity subset only, where STENCIL itself falls to 42.75%. Giving the full-cohort figure is what makes the crossover interpretable: the comparison is not against a model that is uniformly weaker, but against one that stops being helped by retrieval once the available template carries little sequence identity.

Quality features at $K = 1$: the two numbers. Two different operations are reported under this heading and they are not interchangeable. Training a variant without the quality features changes pooled recovery by -0.048 points (95 per cent interval $[-0.164, +0.067]$), which includes zero: the features are not needed to reach this accuracy. Zeroing the same features at inference on the trained model instead costs 2.24 points, from 73.068 to 70.826. The model therefore does use the features when they are present, but recovers what they provide from other signals when it is trained without them. We state both numbers so that "dispensable" is not read as "uninformative".

Complete-case cohort for Table 4. The validation split contains 608 queries. Of these, 606 have a top-1 template, and 526 of those have *both* an exact-superposition top-1 and a Foldseek mapping for the same template, which is what the two mapping columns of the design require; the remaining 80 carry only one of the two mappings and so cannot enter a mapping-source contrast at all. Of the 526, nine have empty evidence under both retrieval-selection conditions and a selected template under

Table 12: Full correspondence corruption series from a separate validation export with a common top-1 template. The dose rows use the 606-query export; the template-free reference uses a 608-query set and is therefore not strictly paired with the other rows. The synchronized fixed-checkpoint intervention values are reported separately in Table 3.

Condition	Queries	Recovery (%)	NLL
Exact correspondence	606	73.193	0.940
5 % permuted	606	70.809	1.071
10 % permuted	606	67.910	1.185
25 % permuted	606	59.736	1.433
50 % permuted	606	50.187	1.689
100 % permuted	606	40.691	1.966
Composition-preserving shuffle	606	40.606	1.963
No template	608	42.8932	1.869

Table 13: Fixed-checkpoint evidence interventions on 608 validation queries. The quality-permutation row is an identity operation at $K = 1$ because there is only one template row to permute; it is listed for completeness and carries no information.

Intervention	Recovery (%)	NLL
Full evidence	73.0683	0.944
Empty evidence	42.8932	1.869
Sequence shuffle	40.573	1.959
Local shift by one residue	41.5080	1.921
Comp.-preserving global shuffle	40.5363	1.964
Random template from top-32	52.008	1.586
Quality features zeroed	70.826	0.994
Quality permutation	73.0683	0.944

exact selection; they are excluded because those rows do not represent a pure change of selection or correspondence source, leaving 517 queries and 90,224 scored residues for the complete-case analysis. The valid cohort and the excluded query identifiers are retained in the released results artifact.

High-recovery subgroup. Restricting the token permutation to the 358 test proteins with recovery at least 0.9 gives a subgroup diagnostic drop of 56.29 points, from 97.52 to 41.23, with 85.5 per cent of that subgroup falling below 0.5 recovery. This cohort is selected using test recovery and is therefore reported only as a subgroup diagnostic; the primary result is the full-test version in Section 6.1.

Coverage features omitted from Table 5. Query coverage separates the high-recovery group with an AUC of 0.863 and template coverage with an AUC of 0.848, both below the sequence-identity and TM-score values reported in the main text.

E Structural self-consistency

Table 14: Fold-back of five sequence conditions through a single structure prediction oracle on 606 validation queries. All five conditions use the same queries and the same alignment procedure. The load-bearing comparison is the model against template-copy: the model's designs fold back closer to the native backbone than the template sequence they were conditioned on.

Sequence source	Mean sc-TM	Median sc-TM	Mean Cα-RMSD	sc-TM > 0.5
Native	0.843	0.925	1.70 Å	92.6 %
Stencil	0.831	0.909	1.82 Å	90.1 %
Template copy	0.795	0.874	2.08 Å	88.1 %
Template-free model	0.787	0.883	2.11 Å	84.5 %
Shuffled evidence	0.707	0.824	2.63 Å	71.3 %

This evaluation uses validation data and a single structure prediction oracle, so it supports the relative ordering rather than absolute foldability. We do not restate recovery as a designability or foldability claim.

F Template redundancy and efficiency

Table 15: Union coverage and mean template quality as a function of the template budget, on the validation split. Additional templates increase coverage marginally while their average quality falls, which is consistent with the small gain from raising the budget beyond one.

K	1	2	4	8	16	32
Mean union coverage	0.901	0.924	0.944	0.962	0.973	0.981
Mean TM-score	0.845	0.826	0.795	0.757	0.710	0.659
Mean identity	0.632	0.577	0.505	0.427	0.352	0.290
Pairwise overlap	–	0.912	0.880	0.844	0.793	0.734

Throughput on a fixed validation subset of 30 proteins and 4,602 residues is 72.8 proteins per second at $K = 1$ against 48.1 at $K = 16$, a factor of 1.51. This measures the cached-evidence forward pass only and excludes retrieval and alignment, which are reported in Appendix B. It is timed on a Tesla T4 as the median of five repeats, and because this subset differs from the 100 length-stratified test queries of Table 8 the two throughput figures are not comparable; only the $K = 1$ versus $K = 16$ ratio within this measurement should be read across template budgets.

G Multi-seed stability, the ensemble diagnostic, and remaining aggregation results

Three independently trained seeds give validation recovery of 73.066, 73.419 and 73.235 per cent, with a sample standard deviation of 0.177 points. The first of these is the canonical checkpoint reported elsewhere in the paper as 73.0683; the 0.0023 difference is reporting precision, not a different model. Pooled paired analysis against the variant without quality features gives a difference of -0.048 points with a 95 per cent confidence interval of $[-0.164, +0.067]$, which spans zero.

Table 16: Equal-weight probability averaging over three seeds, evaluated once on the test set after all model choices were frozen. The single-model values are synchronized with the canonical Table 1 entry; differences are recomputed accordingly. Reported as a mechanistic diagnostic only; excluded from Table 1 because it triples the training budget.

	Single model	Three-seed average	Difference
micro recovery	69.342	70.453	$+1.111$
macro recovery	65.748	66.787	$+1.039$
median recovery	58.350	60.687	$+2.337$
micro NLL	1.0451	0.9724	-0.0727

What the ensemble gain shows. The three-seed average does not raise all three aggregations by the same amount: median recovery gains $+2.34$ points while micro gains $+1.11$ and macro $+1.04$, so the micro–median separation contracts from 10.99 to 9.77 points. That asymmetry is the signature the heterogeneity account predicts. Variance reduction helps the per-protein median, which is pinned down by how widely individual proteins differ, far more than it helps the residue-pooled mean, and the separation shrinks accordingly. It is therefore independent evidence that the separation is a dispersion effect rather than a length-weighting artefact. We still report the single model only. Writing the difference as $\rho(n, a) \cdot \mathrm{CV}(n) \cdot \mathrm{sd}(a)$, where n is the number of scored residues and a the per-protein recovery, Stencil has $\rho = 0.274$, CV $= 0.507$ and sd $= 0.259$, while ProteinMPNN, the non-retrieval reference, has $\rho = 0.402$, the same CV, and sd $= 0.096$. Both sets of statistics are computed on the same CATH 4.2 test cohort and the same coordinate-validity denominator as Table 1, from per-protein records, so the identity reproduces each model's own micro–macro difference (3.59 and 1.95 points respectively). A counterfactual decomposition attributes $+2.80$ points of the excess to performance dispersion and -1.15 points to the weaker length–accuracy correlation.