SHRINKFLOW: LEARNING TO SHRINK PROTEINS FROM PROTEIN LANGUAGE MODEL FEEDBACK

Anonymous authors
Paper under double-blind review

ABSTRACT

Protein shrinking seeks shorter variants of existing proteins, yet experimentally validated source–shortened pairs are scarce and substantial shortening may require coordinated deletions across multiple regions. We introduce SHRINKFLOW, a framework for learning reusable multi-region shrinking policies from protein language model (PLM) feedback. A frozen PLM evaluates complete shortened candidates relative to their source, and the resulting plan preferences are converted into history–action supervision for a conditional discrete interval flow. A feasible action space enforces exact deletion budgets while preserving the identity and order of retained residues. Across general proteins and protein complexes, SHRINKFLOW achieves improved predicted structural and interface preservation relative to recent learning-based shrinking methods. Under shared functional-site constraints, it also preserves the local structural environment around protected catalytic regions more effectively. Controlled studies show that complete-outcome evaluation improves candidate selection over WT-only scoring and that the released policy achieves higher first-output structural quality than a capability-stage reference. Together, these results support reusable, exact-budget protein shrinking from sequence-model feedback without experimentally paired shortened targets.

1 INTRODUCTION

Protein shrinking seeks shorter variants of existing proteins while preserving properties relevant to their use. Reduced sequence length can help satisfy delivery and engineering constraints, as illustrated by engineered miniature CRISPR systems (Xu et al., 2021). The difficulty is not simply achieving a shorter sequence: deleting a region can remove a structural element or functional site, change long-range interactions, and create new sequence neighborhoods at deletion boundaries. Experimental studies show that the effects of insertions and deletions depend strongly on position (Topolska et al., 2025). For substantial shortening, several regions may need to be removed together, making the complete deletion plan a natural unit for evaluating and learning shrinking strategies. Although natural protein sequences are abundant, experimentally validated source–shortened pairs provide limited supervision for this combinatorial task. ProteinGym and MaveDB collect measured sequence perturbations (Notin et al., 2023; Rubin et al., 2025), but these measurements do not directly cover the multi-region deletion plans needed for general protein shrinking. Existing methods consequently rely on alternative learning signals: SCISOR learns deletion operations by reversing synthetic insertions, whereas Raygun reconstructs proteins through a length-controllable representation (Baron et al., 2025; Devkota et al., 2026). PLM-guided search provides another approach by evaluating candidate shortened enzymes directly (Lála et al., 2026). These approaches leave open a complementary question: can evaluations of complete shortened sequences be converted into supervision for a reusable shrinking policy?

We introduce SHRINKFLOW, a framework that turns PLM evaluations of complete shortened sequences into supervision for a reusable multi-region deletion policy. For each source and deletion budget, a frozen PLM evaluates complete deletion plans by comparing the resulting shortened sequences with the source. Plans with higher relative preference receive greater training weight, and each plan is serialized into history–action examples for sequential learning. Proposal correction accounts for how plans enter the finite training pool. A conditional discrete interval flow then learns to generate deletion intervals from the source representation, preceding deletions, and remaining

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.

budget. At inference, the source is encoded once and the policy generates a complete deletion plan within a feasible action space that guarantees exact-budget completion.

Computational evaluations on general proteins and protein complexes show higher predicted structural and interface-preservation metrics for SHRINKFLOW relative to the evaluated learning-based shrinking methods. A family-screened cohort extends the structural comparison beyond the training deletion range, while functional-site experiments distinguish spontaneous annotation retention from local structural preservation under explicit site-protection constraints. Fixed-pool studies evaluate the information provided by complete-outcome scoring, and additional controls compare proposal strategies, checkpoint references, and matched training variants.

Our contributions are:

- We introduce SHRINKFLOW, which converts PLM evaluations of complete shortened sequences into plan-level supervision for reusable, exact-budget multi-region protein shrinking.
- We develop plan-weighted history–action supervision and a conditional discrete interval flow over feasible actions, producing multi-region deletion plans that satisfy exact budgets while preserving retained residue identity and order.
- We present a systematic computational evaluation across general proteins, protein complexes, site-protected enzymes, high-removal stress tests, and eight application case studies, together with controlled analyses of outcome scoring, proposal generation, and training supervision.

2 RELATED WORK

Protein Shrinking. Recent protein-shrinking methods differ in how they learn to shorten an existing sequence. Raygun reconstructs natural proteins through a length-independent representation, enabling miniaturization and other variable-length edits (Devkota et al., 2026). SCISOR learns deletion operations by reversing synthetic insertions (Baron et al., 2025). For enzyme miniaturization, PLM embeddings have also been used to construct energy landscapes and search directly for shortened designs (Lála et al., 2026). Reward-conditioned masked diffusion and adaptive protein tokenization offer further approaches to generating shorter proteins (Rector-Brooks et al., 2025; Dilip et al., 2026). Our focus is on the supervision linking outcome evaluation to a reusable deletion generator: SHRINKFLOW evaluates complete shortened candidates and uses the resulting plan preferences to weight sequential interval training under an exact deletion budget.

Protein Language Models and Design Feedback. Evolutionary covariation models and MSA Transformer learn dependencies from protein families (Hopf et al., 2017; Rao et al., 2021), while sequence PLMs support zero-shot variant assessment (Meier et al., 2021). Tranception, PoET, and PoET-2 incorporate family context for assessment and generation, including indel variants (Notin et al., 2022; Truong Jr & Bepler, 2023; 2026). Structure-aware representations, joint sequence–structure modeling, and biophysical pretraining further expand the information available in protein representations (Su et al., 2024; Hayes et al., 2025; Gelman et al., 2025). Interpretability studies have also linked learned features to functional sites and structural motifs (Simon & Zou, 2025). These representations provide design feedback for comparing sequence variants. In SHRINKFLOW, source–outcome representational consistency serves as a plan-level preference signal, allowing the effects of combined deletions to be evaluated after the shortened sequence is realized.

Discrete Generation and Feedback-Weighted Learning. Flow matching learns generative dynamics through conditional probability paths; discrete formulations commonly parameterize transitions by rates (Lipman et al., 2022; 2024; Gat et al., 2024; Campbell et al., 2024). Related approaches include structured denoising, continuous-time discrete processes, and probability-ratio estimation (Austin et al., 2021; Campbell et al., 2022; Lou et al., 2023). Dirichlet flow matching has been applied to DNA design, while Edit Flows models variable-length edits through insertion, deletion, and substitution (Stark et al., 2024; Havasi et al., 2026). External evaluation signals also guide generation through adaptive conditioning, GFlowNets, energy-based generator matching, and reward-guided diffusion (Brookes et al., 2019; Jain et al., 2022; Woo et al., 2026; Su et al., 2026).

SHRINKFLOW builds on rate-based discrete generation but operates over feasible deletion intervals, coupling complete-plan preferences with history-conditioned supervision under an exact deletion budget.

3 METHOD

SHRINKFLOW constructs supervision from complete-outcome PLM evaluations and learns a conditional generator of deletion intervals. Figure 1 distinguishes the training pipeline from inference: complete plans provide the feedback used during learning, whereas generation makes sequential decisions within a feasibility-constrained action space.

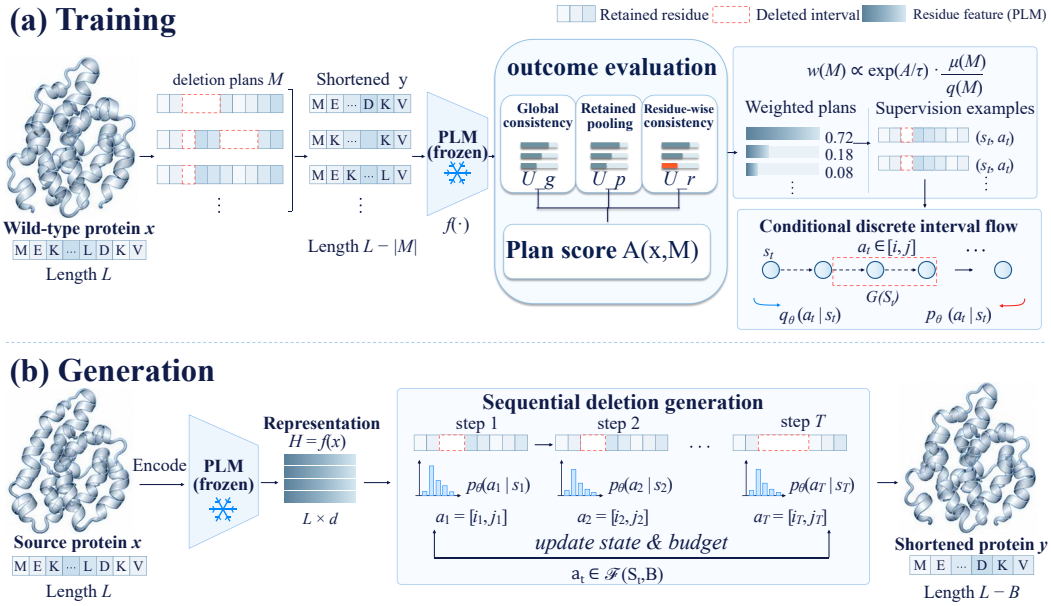

Figure 1: **Complete-outcome feedback and constrained generation. (a) Training:** A frozen PLM encodes the source and each complete shortened sequence. Plan-level preferences determine the sampling of history–action examples for conditional interval-flow training. **(b) Generation:** The source is encoded once; intervals are generated and committed sequentially, updating the deletion history and remaining budget until the target length is reached.

3.1 TASK DEFINITION AND DELETION PLANS

Let $x = (x_1, \ldots, x_L)$ be the source, or wild-type (WT), sequence. A shrinking request removes exactly B residues to produce y of length $L' = L - B$, while preserving the identities and relative order of retained residues. A deletion plan M consists of maximal contiguous intervals in WT coordinates. Each interval has length at least m, successive intervals are separated by at least one retained residue, and their lengths sum to B. We consider requests with $m \le B \le L - 1$. We serialize each plan from right to left. The history h records committed intervals, remaining budget, and a cursor limiting the position of the next interval. The feasible action set $\mathcal{A}(h)$ contains only intervals whose deletion either exhausts the remaining budget or admits a feasible continuation. Committing an interval moves the cursor so that successive maximal deletion intervals remain separated by at least one retained residue. This serialization removes alternative interval orderings of the same deletion mask and separates validity from preference: the action space guarantees a legal completion, while the learned model selects among feasible intervals. For site-protected requests, feasible plans must additionally retain all supplied protected positions. Terminal-only deletions are included as a restricted subset of the same interval-plan representation, while the general action space also permits internal and distributed deletions. Appendix A gives the reference distribution and completion argument.

3.2 Evaluating Complete Shortened Outcomes with a PLM

We obtain plan-level feedback by evaluating the *realized shortened sequence* produced by a complete deletion plan. PLM representational consistency defines relative preferences among plans for the same source and deletion budget. This construction evaluates each candidate after all requested deletions have been applied, without requiring action-level quality labels. A frozen ESM Cambrian model (Candido et al., 2026) separately encodes x and y, giving $H = f(x) \in \mathbb{R}^{L\times d}$ and $H' = f(y) \in \mathbb{R}^{L'\times d}$, where d is the embedding dimension.

Let $r_1, \ldots, r_{L'}$ be the retained WT indices. We denote by $\bar{H}$ the mean over all WT embeddings, by $\bar{H}_R$ the mean of $H_{r_1}, \ldots, H_{r_{L'}}$, and by $\bar{H}'$ the mean over the shortened sequence. With cosine similarity c, the three comparisons are

$$\begin{aligned} U_g(x, M) &= c(\bar{H}, \bar{H}'), \\ U_p(x, M) &= c(\bar{H}_R, \bar{H}'), \\ U_r(x, M) &= \frac{1}{L'} \sum_{k=1}^{L'} c(H_{r_k}, H'_k). \end{aligned} \tag{1}$$

Here, U_g compares the shortened outcome with the entire source, U_p compares pooled embeddings of the same retained residues before and after shortening, and U_r measures residue-wise contextual consistency. The latter two use the exact residue correspondence supplied by the deletion mask. All three scores are computed from PLM representations and deletion-mask correspondence, without task-specific functional annotations.

Raw similarities vary across proteins, deletion budgets, and scoring axes. For each protein–budget condition, independently sampled calibration plans provide the mean and standard deviation of each axis. We standardize the three scores, average them, and clip the result to obtain the complete-plan score $A(x, M)$. The resulting score therefore defines a relative preference within each condition rather than an absolute measure of preservation. Calibration settings are given in Appendix B.

3.3 From Complete-Outcome Evaluation to History–Action Supervision

Complete-outcome feedback assigns preferences to entire deletion plans, while the generator learns interval decisions conditioned on partial histories. We connect these levels by weighting plans before sampling their constituent history–action examples. To specify how plans with different numbers of intervals enter this weighting, we use a span-count-balanced reference measure $\mu_{L,B}$. It assigns equal mass to each feasible interval count and is uniform over legal plans within that count, balancing reference mass across counts despite differences in their numbers of legal plans.

Training plans are drawn from a sequential proposal that samples uniformly from $\mathcal{A}(h)$ at each step. For a plan M_i containing J_i intervals, its unique serialization gives proposal probability q_i as the product of its sequential action probabilities. The finite-pool plan weights are

$$\widetilde{w}_i = \exp(A(x, M_i)/\tau)\, \frac{\mu_{L,B}(M_i)}{q_i}, \qquad w_i = \frac{\widetilde{w}_i}{\sum_k \widetilde{w}_k}. \tag{2}$$

The quality term favors plans with higher complete-outcome scores, while $\mu_{L,B}(M_i)/q_i$ accounts for the difference between the reference measure and the sequential proposal. The parameter τ controls the strength of the quality preference and is distinct from the generation-time rate temperature.

We sample plan i with probability w_i, then sample one of its J_i decisions uniformly. The preceding intervals define history $h_{i,j-1}$, and interval a_{ij} is the target action. This induces

$$\widehat{\mathcal{D}}_{x,B} = \sum_i \frac{w_i}{J_i} \sum_{j=1}^{J_i} \delta_{(h_{i,j-1}, a_{ij})}, \tag{3}$$

where δ is a point mass on a history–action pair. Uniform decision sampling gives each plan total mass w_i, rather than increasing its contribution with its number of intervals. This transfers plan-level preferences to sequential decisions without requiring deletion-level quality labels.

This construction keeps quality supervision at the plan level. No separate quality target is required for each deletion interval: plans receiving larger complete-outcome weights contribute more training

examples, while the $1/J_i$ factor prevents plans with more intervals from receiving additional mass solely because they contain more decisions. The generator therefore learns local history-conditioned actions from preferences assigned to complete shortened sequences.

3.4 Conditional Discrete Interval Flow

Generation alternates between an inner interval flow and an outer deletion process. For a fixed deletion history, the inner flow evolves a candidate interval within the feasible action set. The outer process commits the resulting interval and updates the history, cursor, and remaining budget before generating the next one. We define the inner flow on a local graph whose nodes are legal intervals under the current history and whose edges are feasibility-preserving boundary changes or translations. Graph transitions update the candidate interval while keeping the committed deletion history fixed. A conditional rate network predicts nonnegative transition rates from the current node v to neighboring intervals $b \in \mathcal{N}(v)$, following rate-based discrete generation (Campbell et al., 2022; Gat et al., 2024; Campbell et al., 2024). Its inputs include frozen WT representations, the current deletion mask, requested and remaining budgets, the cursor, and flow time t.

For a history–action pair (h, a^*) sampled from Equation 3, we draw an initial interval v_0 uniformly from $\mathcal{A}(h)$ and connect it to a^* using the deterministic feasible-path construction in Appendix B.1. Write this path as $\mathcal{P} = (v_0, \ldots, v_D)$, where D is the number of graph transitions. At sampled time $t \sim \mathrm{Unif}([0, t_{\max}))$, with $t_{\max} < 1$, the bridge position and target rates are

$$K \sim \mathrm{Binomial}(D, t), \qquad u_t(v_K, b) = \begin{cases} \dfrac{D-K}{1-t}\mathbf{1}\{b = v_{K+1}\}, & K < D, \\ 0, & K = D. \end{cases} \tag{4}$$

As t increases, the bridge shifts probability toward later positions along the constructed path. The target rates guide transitions to the next feasible node on that path, providing directly computable conditional rate supervision (Lipman et al., 2022; 2024).

We train on the empirical history–action distribution using

$$\mathcal{L}_{\text{flow}} = \mathbb{E}\left[\frac{\sum_{b\in\mathcal{N}(v)}\left[r_\theta(v,b) - u_t(v,b)\log r_\theta(v,b)\right]}{1+\sum_{b\in\mathcal{N}(v)} u_t(v,b)}\right]. \tag{5}$$

The expectation is over history–action pairs from $\widehat{\mathcal{D}}_{x,B}$, initial intervals, flow times, and sampled bridge positions. Conditioning arguments of r_θ are omitted for readability. Appendix B.1 derives the conditional optimum of this weighted rate-regression objective.

3.5 Training and Target-Length Generation

Training uses a capability stage followed by a quality-supervised stage. The capability stage pretrains the interval-rate network on history–action examples obtained by decomposing random deletion masks into maximal intervals and serializing them from right to left, without PLM quality labels. Quality-supervised training then introduces frozen WT embeddings and samples history–action examples using the complete-plan weights in Equation 2; detailed datasets, minimum-span settings, and optimization parameters are given in Appendix B. At inference, the source is encoded once and its representation is reused throughout sequential generation. The generator conditions on the committed deletion history and remaining budget without re-encoding intermediate shortened sequences. Each committed interval removes at least m residues and preserves a feasible continuation, so generation terminates exactly at the requested deletion budget. Repeated runs produce multiple complete plans, which can optionally be ranked by complete-outcome evaluation.

4 Experiments

4.1 Experimental Setup

We evaluate ShrinkFlow on General-240 and Complex-150 at 20%, 30%, and 40% sequence removal. The family-screened General-60 cohort is additionally evaluated through 70% removal as a

higher-budget stress test. Functional-site studies examine annotated site retention and local structure under shared protection constraints, while controlled studies compare scoring rules, proposal strategies, checkpoint references, and training supervision. Quality-supervised training uses 250,000 training and 10,000 validation proteins from UniRef50, Swiss-Prot, and CATH (Suzek et al., 2015; The UniProt Consortium, 2025; Dawson et al., 2017), and we compare against SCISOR and Raygun-500 (Baron et al., 2025; Devkota et al., 2026). The primary General and Complex comparisons use the first output returned by each method's native pipeline; Raygun-500 retains its native candidate ranking. General proteins use ESMFold-v1 (Lin et al., 2023), whereas Complex-150 uses empty-MSA Boltz-2 with the interaction partner unchanged (Passaro et al., 2025). Controlled candidate-generation comparisons use matched candidate counts and a common PLM selector. Structural endpoints include full-WT-normalized TM (Zhang & Skolnick, 2005), retained-residue Cα-lDDT (Mariani et al., 2013), fixed-denominator native-contact recovery, and interface-specific measurements. Full protocols and statistical procedures are given in Appendix C.

4.2 Structural and Interface Preservation

On General-240, ShrinkFlow achieves higher mean full-WT TM, retained-residue Cα-lDDT, and native-contact recovery than SCISOR and Raygun-500 (Table 1). Its full-WT TM advantage over SCISOR is approximately 0.11 across 20–40% removal. At 40% removal, full-WT TM is 0.402, compared with 0.294 for SCISOR and 0.236 for Raygun-500; native-contact recovery is 29.31%, compared with 16.13% and 16.23%, respectively.

Table 1: **Structural preservation on General-240.** Results use the first output returned by each method's native pipeline. Bold indicates the highest displayed mean within each removal level. † denotes the midpoint of the lower–upper identification bounds for Raygun correspondence-sensitive metrics.

Removed	Method	Full-WT TM ↑	Cα-lDDT ↑	Contacts (%) ↑
20%	ShrinkFlow	**0.594**	**0.623**	**49.41**
	SCISOR	0.485	0.398	37.32
	Raygun-500	0.308	0.252	23.10†
30%	ShrinkFlow	**0.486**	**0.582**	**38.45**
	SCISOR	0.377	0.308	24.95
	Raygun-500	0.257	0.196	18.07†
40%	ShrinkFlow	**0.402**	**0.547**	**29.31**
	SCISOR	0.294	0.247	16.13
	Raygun-500	0.236	0.192	16.23†

On the same family-screened General-60 cohort, ShrinkFlow has higher mean full-WT TM than SCISOR and Raygun at every evaluated removal level (Figure 2). At 50% removal, the three methods obtain 0.329, 0.233, and 0.200, respectively; at 70%, they obtain 0.193, 0.156, and 0.149. Full-WT normalization incorporates retained-length coverage as well as geometric agreement: for a shortened sequence of length L', the score is bounded by L'/L. On unconstrained single-domain proteins, terminal truncation provides a particularly strong reference: it preserves one contiguous interior segment without introducing internal deletion junctions, and in the matched General-240 comparison it achieves higher whole-chain structural scores than ShrinkFlow (Appendix F.3). Under shared site-protection constraints, however, terminal truncation is not always feasible: a valid terminal-only plan exists for 28 of 36 proteins at 20% removal and for 12 of 36 at 70%, while ShrinkFlow produces valid exact-budget plans under the same constraints for all 36 proteins.

On Complex-150, ShrinkFlow achieves higher predicted native interface-contact recovery and lower partner-frame RMSD than SCISOR and Raygun-500 at all three removal levels. At 20% removal, interface-residue identity is close to SCISOR (80.8% versus 79.2%), while native interface-contact recovery is 17.22% versus 7.63%. At 40%, contact recovery remains higher at 10.64%, compared with 3.42% for SCISOR and 2.10% for Raygun-500. Absolute interface recovery remains limited, and differences from SCISOR are smaller on the family-screened Complex-30 cohort.

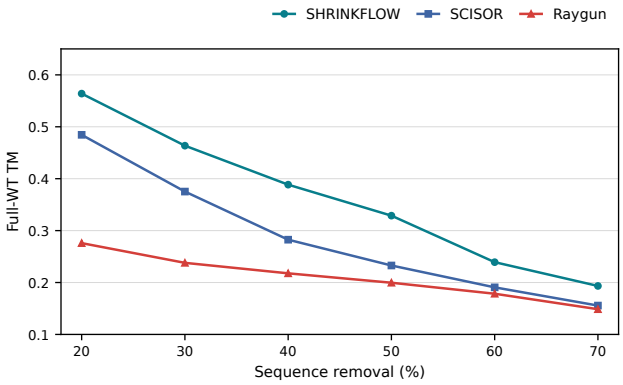

Figure 2: **Structural preservation across increasing deletion budgets on General-60.** Mean full-WT TM is shown for the same family-screened 60-protein cohort from 20% to 70% sequence removal. The score is bounded by the shortened-to-WT length ratio. Results use the first output returned by each method's evaluation pipeline. Raygun results at 60% and 70% combine native successes with a short-input repair for two and ten failed conditions, respectively. Native delivery counts, native-only results, and repair details are reported in Appendix D.3.

Table 2: **Interface preservation on Complex-150.** Identity is retained interface-residue identity; RMSD is measured in the unchanged partner's frame. †: identification-bound midpoint.

Removed	Method	Identity (%) ↑	Contacts (%) ↑	RMSD (Å) ↓
20%	SHRINKFLOW	**80.8**	**17.22**†	**24.39**
	SCISOR	79.2	7.63†	28.68
	Raygun-500	51.1†	2.39†	29.57
30%	SHRINKFLOW	69.02	**12.13**†	**25.03**
	SCISOR	**69.23**	5.39†	28.40
	Raygun-500	46.16†	2.35†	28.78
40%	SHRINKFLOW	**64.5**	**10.64**†	**25.66**
	SCISOR	59.4	3.42†	29.06
	Raygun-500	41.4†	2.10†	28.42

4.3 STRUCTURAL PRESERVATION UNDER FUNCTIONAL-SITE CONSTRAINTS

We evaluate shrinking under shared site-protection constraints on 20 enzymes at 20% and 40% removal. SHRINKFLOW and SCISOR receive the same catalytic and specified cofactor/metal-contact positions as non-deletable residues, and both retain all supplied protected positions. Each method generates eight candidates per condition, from which the same annotation-blind PLM scorer selects one. Selected candidates are refolded using the same single-chain, empty-MSA Boltz-2 configuration. SHRINKFLOW achieves higher fixed-denominator local Cα scores and full-WT TM, together with lower catalytic-geometry MAE than SCISOR (Table 3). Averaged across the two budgets, the local Cα score is 0.485 for SHRINKFLOW and 0.267 for SCISOR, with SHRINKFLOW higher on 19 of 20 proteins. The separation becomes larger at 40% removal, where the local Cα scores are 0.475 and 0.180, respectively. The same trend is also observed in the first-output control without candidate reranking, showing that the difference is already present in the generated candidates. The two methods also produce markedly different deletion patterns: SHRINKFLOW typically concentrates removal into a small number of long intervals, whereas SCISOR distributes the same deletion budget across many short deletions. This difference is consistent with the complementary unprotected-site results, where SCISOR retains more M-CSA catalytic residues, while the gap is smaller on the UniProt catalytic/binding union. When catalytic and cofactor/metal-contact positions are supplied as explicit constraints, both methods retain all of them and the comparison instead concerns the surrounding structural environment.

Table 3: **Structural preservation under shared site protection.** Values are means over 20 enzymes after common best-of-eight PLM selection and matched Boltz-2 prediction. Local Cα uses a fixed WT denominator around catalytic sites. Catalytic-geometry MAE uses the experimental WT reference.

Removed	Method	Local Cα score ↑	Full-WT TM ↑	Catalytic MAE (Å) ↓
20%	SHRINKFLOW	**0.494**	**0.399**	**3.954**
	SCISOR	0.355	0.345	5.110
40%	SHRINKFLOW	**0.475**	**0.327**	**4.648**
	SCISOR	0.180	0.251	7.090

4.4 COMPLETE-OUTCOME EVALUATION AND POLICY ANALYSIS

We first examine the value of complete-outcome evaluation for candidate selection. On identical General-240 candidate pools at 30% removal, complete-outcome scoring increases full-WT TM from 0.476 to 0.576 and native-contact recovery from 35.99% to 49.79% relative to WT-only scoring (Table 4). The effect transfers to a second PLM: on Enzyme-120, both ESMC and ESM-2 selection improve mean full-WT TM over the first candidate at all three removal levels, and at 20% removal the fraction of proteins retaining all annotated catalytic positions increases from 37.82% to 53.78% and 54.62%, respectively. In a separate matched-morphology study, Joint scoring also improves over WT-only evaluation but shows no clear selection advantage over three additive per-span constructions.

Table 4: **Complete-outcome scoring on General-240 at 30% removal.** Both scoring rules select from identical candidate pools.

Scoring rule	Full-WT TM ↑	Cα-lDDT ↑	Contacts (%) ↑
WT-only	0.476	0.545	35.99
Complete outcome	**0.576**	**0.755**	**49.79**

We next compare proposal strategies under a common selector. On the 24-protein development panel, SHRINKFLOW, reference μ sampling, and PLM-guided local search use matched candidate budgets and the same complete-outcome scorer. SHRINKFLOW yields higher selected TM and native-contact recovery at all three removal levels (Table 5); at 30%, contact recovery is 58.5%, compared with 48.6% and 40.0%. A separate protected 36-protein panel provides a stronger legal-sampling comparison, where common $K = 32$ selection does not clearly separate SHRINKFLOW from sequential-uniform sampling in native-contact recovery.

Table 5: **Candidate quality under a shared PLM scorer.** TM is full-WT-normalized and contact recovery is reported as a percentage.

	20% removed		30% removed		40% removed	
Strategy	TM ↑	Contacts (%) ↑	TM ↑	Contacts (%) ↑	TM ↑	Contacts (%) ↑
SHRINKFLOW	**0.619**	**60.9**	**0.555**	**58.5**	**0.461**	**47.7**
Reference μ	0.588	60.1	0.498	48.6	0.389	32.7
PLM-local	0.558	52.3	0.455	40.0	0.393	28.8

Finally, we examine how the learned policy changes beyond the capability-stage reference. On a fixed 30-protein panel with the same minimum-span setting, the released policy has 0.032 higher mean native-contact recovery on first unselected outputs (Appendix F.5). The difference becomes smaller after common best-of-eight selection, as the same selector can recover strong candidates from both proposal distributions. The quality signal also changes the training weights substantially: across deletion budgets, the total-variation distance between Full and NoReward plan weights is 0.27–0.29, and the corresponding Full–WT-only distance is 0.40–0.42. Matched 20,000-update training controls give smaller and less uniform structural differences across individual supervision variants; complete results are reported in Appendix E.

4.5 CASE STUDIES

We further evaluate SHRINKFLOW on eight miniaturization targets reported in the Raygun study (Devkota et al., 2026): HBB, CCR1, lacZ, mTOR, eGFP, mCherry, TurboID, and SpCas9. For each target, all methods use the same WT sequence and target length, with eight candidates refolded under a common AlphaFold 3 protocol (Abramson et al., 2024). Across these targets, SHRINKFLOW achieves the highest macro-average on all five structural and confidence metrics in Table 6, and the highest retained-residue Cα-lDDT and native-contact recovery on all eight targets. Full per-target results are reported in Appendix G.

Table 6: **Average performance on eight Raygun miniaturization targets.** Each target contributes equally; values are averaged over eight candidates per target. Bold indicates the highest mean.

Method	pLDDT ↑	pTM ↑	Full-WT TM ↑	Cα-lDDT ↑	Contacts ↑
SHRINKFLOW	**78.36**	**0.748**	**0.755**	**0.895**	**0.698**
SCISOR	63.79	0.609	0.651	0.635	0.530
Raygun	48.81	0.392	0.454	0.497	0.338

SpCas9 provides a representative complex-valued example. At 28.9% removal, SHRINKFLOW achieves mean full-WT TM 0.596, Cα-lDDT 0.912, and ipTM 0.806 across eight candidates, compared with 0.387/0.411/0.671 for SCISOR and 0.186/0.249/0.186 for Raygun.

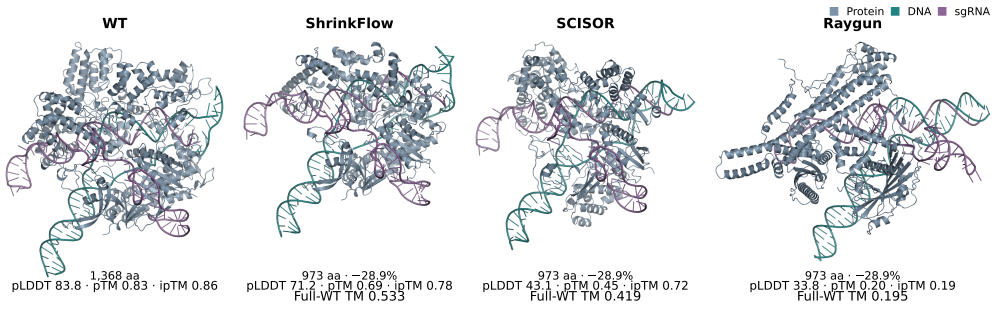

Figure 3: **AF3 refolding of SpCas9 at 28.9% sequence removal.** The WT complex and candidate `00`, the first returned 973-residue design from SHRINKFLOW, SCISOR, and Raygun, are shown under the same alignment, camera, and scale. Displayed metrics refer to the individual structures shown. pLDDT, pTM, and ipTM are AF3 confidence metrics; Full-WT TM uses the corresponding same-protocol AF3 WT prediction as reference.

5 CONCLUSION

We presented SHRINKFLOW, a framework for learning reusable, exact-budget multi-region protein shrinking policies from PLM evaluations of complete shortened sequences. SHRINKFLOW converts plan-level preferences into history–action supervision and learns a feasibility-constrained discrete interval flow for sequential deletion generation. Across general proteins, protein complexes, and site-protected enzymes, SHRINKFLOW achieves higher predicted structural preservation than the evaluated learning-based shrinking methods, including improved interface-contact recovery and better preservation of local catalytic structure under shared constraints. Controlled studies further show that complete-outcome evaluation improves candidate selection and that the learned policy produces stronger candidates before reranking. The interval-plan formulation supports both terminal-only and internal multi-region deletions, allowing the same framework to accommodate different length and site constraints. These results are based on sequence-model feedback and predicted structures, leaving direct activity, stability, and binding measurements for future validation. Incorporating structure-aware, task-specific, or experimentally measured feedback into the same plan-level supervision would allow the shrinking policy to be refined toward the function of interest over successive design rounds.

REPRODUCIBILITY STATEMENT

Appendices A–H document the deletion-plan representation, training objective, model and generation settings, dataset construction, baseline protocols, evaluation metrics, and statistical procedures used in our experiments. The supplementary material provides the SHRINKFLOW implementation, paper-scale configurations, validation tests, and documentation for training, generation, and PLM-consistency evaluation. Full reproduction of the reported structural benchmarks additionally requires the external datasets, model checkpoints, and third-party structure-prediction resources described in the appendices.

AI USE STATEMENT

Generative AI tools were used to assist with drafting and revising manuscript text, literature retrieval, research ideation, software implementation, experiment planning, analysis scripts, and result checking. The authors reviewed the retrieved literature, verified the experimental results and citations, and take full responsibility for the scientific claims and final content of the manuscript.

REFERENCES

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

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.

Ethan Baron, Alan N Amin, Ruben Weitzman, Debora Marks, and Andrew Gordon Wilson. A diffusion model to shrink proteins while maintaining their function. *arXiv preprint arXiv:2511.07390*, 2025.

David Brookes, Hahnbeom Park, and Jennifer Listgarten. Conditioning by adaptive sampling for robust design. In *International conference on machine learning*, pp. 773–782. PMLR, 2019.

Andrew Campbell, Joe Benton, Valentin De Bortoli, Thomas Rainforth, George Deligiannidis, and Arnaud Doucet. A continuous time framework for discrete denoising models. *Advances in Neural Information Processing Systems*, 35:28266–28279, 2022.

Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. *arXiv preprint arXiv:2402.04997*, 2024.

Salvatore Candido, Thomas Hayes, Alexander Derry, Roshan Rao, Zeming Lin, Robert Verkuil, Bryan Z Wu, Jin Sub Lee, Elise S Bruguera, Jehan A Keval, et al. Language modeling materializes a world model of protein biology. *bioRxiv*, pp. 2026–06, 2026.

Natalie L Dawson, Tony E Lewis, Sayoni Das, Jonathan G Lees, David Lee, Paul Ashford, Christine A Orengo, and Ian Sillitoe. Cath: an expanded resource to predict protein function through structure and sequence. *Nucleic acids research*, 45(D1):D289–D295, 2017.

Kapil Devkota, Daichi Shonai, Joey Mao, Young Su Ko, Wei Wang, Scott Soderling, and Rohit Singh. Miniaturizing and modifying natural proteins with raygun. *Nature*, pp. 1–9, 2026.

Rohit Dilip, Ayush Varshney, and David Van Valen. Adaptive protein tokenization. *arXiv preprint arXiv:2602.06418*, 2026.

Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky TQ Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete flow matching. *Advances in Neural Information Processing Systems*, 37: 133345–133385, 2024.

Sam Gelman, Bryce Johnson, Chase R Freschlin, Arnav Sharma, Sameer D'Costa, John Peters, Anthony Gitter, and Philip A Romero. Biophysics-based protein language models for protein engineering. *Nature Methods*, 22(9):1868–1879, 2025.

Marton Havasi, Brian Karrer, Itai Gat, and Ricky TQ Chen. Edit flows: Variable length discrete flow matching with sequence-level edit operations. *Advances in Neural Information Processing Systems*, 38:139037–139070, 2026.

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

Thomas A Hopf, John B Ingraham, Frank J Poelwijk, Charlotta PI Schärfe, Michael Springer, Chris Sander, and Debora S Marks. Mutation effects predicted from sequence co-variation. *Nature biotechnology*, 35(2):128–135, 2017.

Moksh Jain, Emmanuel Bengio, Alex Hernandez-Garcia, Jarrid Rector-Brooks, Bonaventure FP Dossou, Chanakya Ajit Ekbote, Jie Fu, Tianyu Zhang, Michael Kilgour, Dinghuai Zhang, et al. Biological sequence design with gflownets. In *International conference on machine learning*, pp. 9786–9801. PMLR, 2022.

Jakub Lála, Harsh Agrawal, Fanfei Dong, Jude Wells, and Stefano Angioletti-Uberti. An energy landscape approach to miniaturizing enzymes using protein language model embeddings. *bioRxiv*, pp. 2026–03, 2026.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

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

Yaron Lipman, Marton Havasi, Peter Holderrieth, Neta Shaul, Matt Le, Brian Karrer, Ricky TQ Chen, David Lopez-Paz, Heli Ben-Hamu, and Itai Gat. Flow matching guide and code. *arXiv preprint arXiv:2412.06264*, 2024.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. *arXiv preprint arXiv:2310.16834*, 2023.

Valerio Mariani, Marco Biasini, Alessandro Barbato, and Torsten Schwede. lddt: a local superposition-free score for comparing protein structures and models using distance difference tests. *Bioinformatics*, 29(21):2722–2728, 2013.

Joshua Meier, Roshan Rao, Robert Verkuil, Jason Liu, Tom Sercu, and Alex Rives. Language models enable zero-shot prediction of the effects of mutations on protein function. *Advances in neural information processing systems*, 34:29287–29303, 2021.

Pascal Notin, Mafalda Dias, Jonathan Frazer, Javier Marchena-Hurtado, Aidan N Gomez, Debora Marks, and Yarin Gal. Tranception: protein fitness prediction with autoregressive transformers and inference-time retrieval. In *International Conference on Machine Learning*, pp. 16990–17017. PMLR, 2022.

Pascal Notin, Aaron Kollasch, Daniel Ritter, Lood Van Niekerk, Steffanie Paul, Han Spinner, Nathan Rollins, Ada Shaw, Rose Orenbuch, Ruben Weitzman, et al. Proteingym: Large-scale benchmarks for protein fitness prediction and design. *Advances in neural information processing systems*, 36: 64331–64379, 2023.

Saro Passaro, Gabriele Corso, Jeremy Wohlwend, Mateo Reveiz, Stephan Thaler, Vignesh Ram Somnath, Noah Getz, Tally Portnoi, Julien Roy, Hannes Stark, et al. Boltz-2: Towards accurate and efficient binding affinity prediction. *BioRxiv*, 2025.

Roshan M Rao, Jason Liu, Robert Verkuil, Joshua Meier, John Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives. Msa transformer. In *International conference on machine learning*, pp. 8844–8856. PMLR, 2021.

Jarrid Rector-Brooks, Mohsin Hasan, Zhangzhi Peng, Chenghao Liu, Sarthak Mittal, Nouha Dziri, Michael Bronstein, Pranam Chatterjee, Alexander Tong, and Joey Bose. Steering masked discrete diffusion models via discrete denoising posterior prediction. In *International Conference on Learning Representations*, volume 2025, pp. 25383–25414, 2025.

Alan F Rubin, Jeremy Stone, Aisha Haley Bianchi, Benjamin J Capodanno, Estelle Y Da, Mafalda Dias, Daniel Esposito, Jonathan Frazer, Yunfan Fu, Sally B Grindstaff, et al. Mavedb 2024: a curated community database with over seven million variant effects from multiplexed functional assays. *Genome biology*, 26(1):13, 2025.

Elana Simon and James Zou. Interplm: discovering interpretable features in protein language models via sparse autoencoders. *Nature methods*, 22(10):2107–2117, 2025.

Hannes Stark, Bowen Jing, Chenyu Wang, Gabriele Corso, Bonnie Berger, Regina Barzilay, and Tommi Jaakkola. Dirichlet flow matching with applications to dna sequence design. *arXiv preprint arXiv:2402.05841*, 2024.

Martin Steinegger and Johannes Söding. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028, 2017.

Jin Su, Chenchen Han, Yuyang Zhou, Junjie Shan, Xibin Zhou, and Fajie Yuan. Saprot: Protein language modeling with structure-aware vocabulary. In *International Conference on Learning Representations*, volume 2024, pp. 6987–7009, 2024.

Xingyu Su, Xiner Li, Masatoshi Uehara, Sunwoo Kim, Yulai Zhao, Gabriele Scalia, Ehsan Hajiramezanali, Tommaso Biancalani, Degui Zhi, and Shuiwang Ji. Iterative distillation for reward-guided fine-tuning of diffusion models in biomolecular design. In *International Conference on Learning Representations*, volume 2026, pp. 139443–139469, 2026.

Baris E Suzek, Yuqi Wang, Hongzhan Huang, Peter B McGarvey, Cathy H Wu, and the UniProt Consortium. Uniref clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*, 31(6):926–932, 2015.

The UniProt Consortium. Uniprot: the universal protein knowledgebase in 2025. *Nucleic acids research*, 53(D1):D609–D617, 2025.

Magdalena Topolska, Antoni Beltran, and Ben Lehner. Deep indel mutagenesis reveals the impact of amino acid insertions and deletions on protein stability and function. *Nature communications*, 16(1):2617, 2025.

Timothy Truong Jr and Tristan Bepler. Poet: A generative model of protein families as sequences-of-sequences. *Advances in Neural Information Processing Systems*, 36:77379–77415, 2023.

Timothy Truong Jr and Tristan Bepler. Understanding protein function with a multimodal retrieval-augmented foundation model. *Advances in Neural Information Processing Systems*, 38:53167–53222, 2026.

Dongyeop Woo, Minsu Kim, Minkyu Kim, Kiyoung Seong, and Sungsoo Ahn. Energy-based generator matching: A neural sampler for general state space. *Advances in Neural Information Processing Systems*, 38:110247–110289, 2026.

Xiaoshu Xu, Augustine Chemparathy, Leiping Zeng, Hannah R Kempton, Stephen Shang, Muneaki Nakamura, and Lei S Qi. Engineered miniature crispr-cas system for mammalian genome regulation and editing. *Molecular cell*, 81(20):4333–4345, 2021.

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.

A Plan Representation and Feasible Completion

A deletion mask has a unique decomposition into maximal contiguous intervals. Right-to-left serialization fixes their order. At each history, the next interval lies to the left of the cursor, has length at least m, and either exhausts the remaining budget or leaves at least one feasible continuation. After committing an interval, the cursor moves past one retained position, keeping successive maximal deletion intervals separated.

Completion. For an initially feasible request, every permitted action preserves the existence of a valid completion. Each action consumes at least m residues, so no more than $\lfloor B/m \rfloor$ intervals can be committed. Whenever the remaining budget is positive, at least one feasible continuation exists, and generation terminates when the remaining budget reaches zero. The resulting intervals are disjoint and separated, and their lengths sum exactly to B. Exact-budget completion therefore follows from the feasible action support.

Span-count-balanced reference. Let $J_{\max} = \min(\lfloor B/m \rfloor, L - B + 1)$ be the maximum feasible number of deletion intervals. For a fixed interval count J, write each interval length as $m + y_j$, with $y_j \geq 0$. Distributing the excess deleted length and placing the retained gaps gives

$$N_J = \binom{B - (m-1)J - 1}{J - 1} \binom{L - B + 1}{J}. \tag{6}$$

The corresponding reference measure is

$$\mu_{L,B}(M) = \frac{1}{J_{\max} N_{J(M)}}. \tag{7}$$

It is uniform over feasible interval counts and uniform over plans conditional on a count.

The training proposal instead samples uniformly from the legal action set at each history, giving proposal probability $q(M) = \prod_j |\mathcal{A}(h_{j-1})|^{-1}$. Sequentially uniform actions do not in general induce a uniform distribution over complete plans conditional on interval count, so $\mu_{L,B}$ and q need not coincide. Both enter the plan weighting in Equation 2. Repeated plans retain their multiplicity in the finite proposal pool.

B Architecture, Training, and Numerical Generation

The interval-rate network combines frozen WT residue embeddings with the current deletion history, cursor, requested and remaining budgets, and flow time. The primary quality-supervised configuration is summarized in Table 7; controlled training variants are described separately in Appendix E.

Two training stages. Training proceeds in two stages. The capability stage initializes legal multi-interval generation without PLM quality supervision. It draws uniform subsets of B deleted positions, decomposes each mask into maximal right-to-left intervals, and samples a prefix uniformly for the interval-rate loss. This stage uses minimum interval length one, removal fractions of 10%, 20%, 30%, and 40%, and a zero PLM input tensor. It is trained on 200,000 proteins with 10,000 validation proteins for 50,000 updates on four ranks.

The subsequent quality-supervised stage introduces frozen WT embeddings, increases the minimum interval length to three, and uses the complete-plan preferences from Equation 2. The primary model is trained at removal fractions 20%, 30%, and 40%. Matched training-supervision controls under the primary proposal-corrected recipe are reported in Appendix E.4.

Calibration and weighting. For each protein–budget condition, eight independently drawn calibration plans provide the population mean and standard deviation for each of the three PLM scoring axes. The standard deviation is floored at 10^{-4}. We standardize the three axes, average them, and clip the resulting score to absolute value eight. This quality term is combined with the proposal correction μ/q to form the finite-pool weights in Equation 2. The raw three-axis mean used in the functional-site selection experiments is reported separately from this calibrated score.

Table 7: **Model and primary quality-supervised training settings.**

Setting	Value
Frozen PLM	ESMC 600M; residue dimension 1,152
Student width / token-mixing layers	512 / 8
Attention heads / dropout	8 / 0.1
Representation channels	8
Training / validation proteins	250,000 / 10,000
Training removal fractions	0.2, 0.3, 0.4
Proposal / calibration plans per condition	32 / 8
Calibration SD floor	10^{-4}; population SD
Absolute advantage clip / target temperature	8 / 1
Optimizer	AdamW
Learning rate / weight decay	3×10^{-5} / 0.01
Primary optimizer updates	20,000
Generation steps / rate temperature	32 / 3
Minimum deletion-interval length	3

B.1 Weighting Induced by the Empirical Objective

Sampling plan i with probability w_i and then sampling one of its J_i decision positions uniformly gives

$$\mathcal{L} = \sum_i \frac{w_i}{J_i} \sum_{j=1}^{J_i} \mathbb{E}[\ell_\theta(h_{i,j-1}, a_{ij})] \,. \tag{8}$$

Each plan therefore contributes the average of its decision losses. Without the $1/J_i$ factor, plans containing more intervals would contribute proportionally more decision losses. The objective transfers complete-plan preferences to their constituent sequential decisions without requiring independently measured action-level quality labels.

The exit-rate normalization in Equation 5 also changes the conditional optimum. Let $s = (x, B, h, v, t)$ denote the network-visible state and let $\omega = (1 + \sum_b u_b)^{-1}$ depend on the sampled bridge target. Ignoring the numerical rate floor, the conditional edge objective is $\mathbb{E}[\omega(r_b - u_b \log r_b) \mid s]$. At a positive interior optimum,

$$r_b^*(s) = \frac{\mathbb{E}[\omega u_b \mid s]}{\mathbb{E}[\omega \mid s]}. \tag{9}$$

Because ω depends on the sampled target, this optimum generally differs from $\mathbb{E}[u_b \mid s]$. Equation 5 is therefore a weighted rate-regression objective. Exact-budget completion follows separately from the feasible action support established in Appendix A.

Factorized path construction. For the primary configuration with minimum interval length three, we draw an initial legal interval v_0 for each sampled target interval a^* and construct a deterministic path in the current factorized action graph. The path first lifts the source interval to the terminal-length layer, shifts its start position within that layer toward the target, and then follows the target lift path in reverse until reaching a^*. Displacements use strides in $\{1, 2, 3, 4, 8, 16, 32\}$. For each nonzero lift, we reserve a final move of three residues, the minimum span length, and greedily decompose the preceding distance using the largest available stride that does not exceed the remaining distance; translations use the same greedy rule without this reservation. The construction does not enumerate alternative paths or use random tie-breaking.

Both endpoints are required to lie in the canonical legal support for the current history, remaining budget, and cursor. After construction, every successive pair of intervals is verified to be an edge of the current legal factorized action graph. If endpoint validation fails, a required displacement cannot be represented, an intermediate transition is not a legal edge, or the constructed path does not terminate at a^*, the implementation raises an exception rather than resampling or falling back to an alternative path.

Bridge and numerical settings. Training samples $t \sim \mathrm{Unif}([0, 0.98))$ and uses a floor of 10^{-8} inside the rate logarithm. The interval graph uses boundary and translation strides in

$\{1, 2, 3, 4, 8, 16, 32\}$, giving at most 42 outgoing neighbors before feasibility filtering. Initial source intervals are sampled uniformly using counts of legal starting positions for each interval length.

Generation uses 32 midpoint integration steps. With step size $\Delta t = 1/32$ and total exit rate λ, the probability of remaining at the current node is $\exp(-\lambda \Delta t)$; otherwise, a neighboring interval is sampled in proportion to its predicted transition rate. The generation-time rate temperature modifies relative transition rates while preserving their sum, and integration proceeds to flow time one. This rate temperature is distinct from the target temperature τ used in complete-plan weighting.

C Datasets, Evaluation Protocols, and Statistics

Evaluation protocols vary across experiments, particularly in whether outputs are taken directly or selected from a candidate pool. Table 8 summarizes the main settings.

Table 8: **Evaluation protocols across experiments.**

Study	Evaluation protocol
General-240 / Complex-150	First native output without shared reranking.
Fixed-pool scoring	Selection from identical General-240 candidate pools using different scoring rules.
Shared-scorer analysis	A common three-axis PLM scorer selects from up to 32 candidates on 24 development proteins.
Unprotected site retention	A common PLM scorer selects one of eight native candidates; functional annotations are used only for evaluation.
Protected-site structure	ShrinkFlow and SCISOR use shared site-protection constraints on 20 enzymes. The main comparison uses a common PLM selector over eight candidates; additional controls evaluate first outputs without reranking ($K = 1$).
Raygun case-study targets	Eight evaluated candidates per method using the same WT sequence and target length within each case; all structures use a common AlphaFold 3 refolding protocol. SpCas9 is evaluated as a protein–DNA–RNA complex, while the remaining targets are single-chain proteins.
Matched supervision controls	Full, WT-only, NoReward, and Shuffled models are evaluated on fresh unselected outputs under the primary $T = 3$ setting, with $T = 1$ reported as a sensitivity analysis.

Primary predictors and native baselines. General-240 and the family-screened General-60 cohort use ESMFold-v1 with three recycles, chunk size 64, half-precision ESM components, and evaluation mode. Complex-150 uses empty-MSA Boltz-2 with three recycles, 200 sampling steps, and one sample while keeping the interaction partner unchanged. The primary ShrinkFlow configuration uses rate temperature three and 32 integration steps. SCISOR uses its native stochastic deletion procedure at temperature one. Raygun-500 generates 500 candidates with argmax decoding, noise one, and one cycle, and returns 50 outputs ranked by adjusted PLM log-score. The case-study suite uses AlphaFold 3 under a matched refolding protocol across methods; all WT-relative case-study metrics use the corresponding same-protocol AF3 WT prediction as reference.

Data sources and family screening. Structural references combine CATH-domain records and curated PDB chains. Complex records identify the target chain, unchanged interaction partner, and supporting biological assembly. Family-screened structural evaluations use 60 general proteins and 30 complexes. Eligibility and family separation are checked before generation using MMseqs2 with 30% sequence identity, 80% query coverage, and within-panel family checks (Steinegger & Söding, 2017). Functional-site datasets are described in Appendix D.4.

Structural endpoints. Full-WT TM uses the complete canonical WT target length L for normalization. For a shortened target of actual length L', at most L' residue pairs can contribute to the TM-score sum, and each contribution is at most one. Consequently, full-WT TM is bounded by L'/L, equal to $(L - B)/L$ for an exact-budget deletion request. Comparisons across removal budgets therefore reflect both retained-length coverage and geometric agreement. Exact deletion masks provide retained-residue correspondence for ShrinkFlow and SCISOR. For general proteins, native contacts are WT Cα pairs separated by at least three sequence positions and within

8 Å. A contact is recovered only when both endpoints remain mapped and their shortened-structure distance remains below 8 Å. All native contacts remain in the denominator, so deletion of either endpoint counts as non-recovery. Retained-residue Cα-lDDT uses reference distances below 15 Å with thresholds of 0.5, 1, 2, and 4 Å. In the protected-site experiment, the fixed-denominator local score additionally assigns zero to reference pairs with a deleted endpoint.

Complex endpoints and evaluable counts. Native inter-chain contacts are heavy-atom residue pairs within 6 Å. Contact recovery uses all native contacts in the denominator. Interface identity measures retention of the corresponding WT amino acid at reference interface positions. Partner-frame RMSD first aligns the unchanged interaction partner and then measures target-interface Cα displacement without refitting the shortened target. Interface identity is evaluable for 149 targets, whereas contact recovery uses all 150, including identification bounds for one unresolved target. At 20%, 30%, and 40% removal, partner-frame RMSD is evaluable for 149/148/147 SHRINKFLOW targets, 149/149/149 SCISOR targets, and 149/148/148 Raygun targets. The unchanged partner is supplied to structure prediction but not to the primary shrinking generator.

Correspondence uncertainty. For Raygun, forward/backward dynamic programming over optimal unit-edit alignments identifies stable residue mappings, certain deletions, and ambiguous correspondences. Ambiguity induces lower–upper identification bounds for site- and contact-based endpoints. Tables display the arithmetic midpoint of these bounds, marked †. Boldface indicates the largest displayed mean within each comparison.

Statistical units. Primary structural effects use 10,000 paired protein-level bootstrap draws. Family-screened effects use paired family resampling, with each protein representing a distinct retained family component. In functional-site studies, the two deletion budgets are averaged within each protein for the reported combined contrast, and the two 20-protein panels are analyzed separately. In the matched supervision study, the eight fresh unselected outputs are first averaged within each protein–budget–model condition, followed by averaging across the three matched training seeds. Paired uncertainty is then estimated by 10,000 protein-level bootstrap resamples. Temperature comparisons use the common evaluable protein set. Candidates from the same protein are therefore not treated as independent statistical units.

D EXTENDED STRUCTURAL AND FUNCTIONAL EVALUATION

D.1 INTERMEDIATE-BUDGET STRUCTURAL RESULTS

We provide paired uncertainty estimates for the primary structural comparisons. The 30% results below reproduce the corresponding main-table entries using the same cohorts and native-output protocols.

Table 9: **General-240 at 30% removal.** †: identification-bound midpoint.

Method	Full-WT TM ↑	Cα-lDDT ↑	Contacts (%) ↑
SHRINKFLOW	**0.486**	**0.582**	**38.45**
SCISOR	0.377	0.308	24.95
Raygun-500	0.257	0.196	18.07†

On General-240, the paired full-WT TM differences from SCISOR at 20%, 30%, and 40% removal are $+0.1092$ (95% CI $[0.0883, 0.1308]$), $+0.1090$ ($[0.0907, 0.1272]$), and $+0.1076$ ($[0.0925, 0.1226]$), respectively. The corresponding retained-residue lDDT differences are $+0.2246$, $+0.2737$, and $+0.3007$, with intervals $[0.2007, 0.2473]$, $[0.2500, 0.2968]$, and $[0.2770, 0.3256]$.

D.2 FAMILY-SCREENED STRUCTURAL EVALUATION

The separately screened cohorts contain 60 general proteins and 30 complexes. General-60 is evaluated at six removal levels from 20% to 70%, using the same proteins throughout. Complex-30 retains the original 20%, 30%, and 40% protocol. On General-60, SHRINKFLOW has higher mean full-WT

Table 10: **Complex-150 at 30% removal.** RMSD is measured in the unchanged partner's frame; † marks identification-bound midpoints.

Method	Identity (%) ↑	Contacts (%) ↑	RMSD (Å) ↓
SHRINKFLOW	69.02	**12.13**†	**25.03**
SCISOR	**69.23**	5.39†	28.40
Raygun-500	46.16†	2.35†	28.78

TM than SCISOR and Raygun across the evaluated budgets. On Complex-30, the corresponding differences from SCISOR are substantially smaller.

Table 11: **Full-WT TM on family-screened cohorts.** Bold compares methods within each row. ‡: native Raygun successes supplemented by the short-input repair described in Appendix D.3.

Task	n	Removed	SHRINKFLOW ↑	SCISOR ↑	Raygun-500 ↑
General	60	20%	**0.564**	0.485	0.276
General	60	30%	**0.464**	0.375	0.238
General	60	40%	**0.389**	0.283	0.218
General	60	50%	**0.329**	0.233	0.200
General	60	60%	**0.239**	0.191	0.178‡
General	60	70%	**0.193**	0.156	0.149‡
Complex	30	20%	0.417	**0.419**	0.281
Complex	30	30%	**0.357**	0.350	0.261
Complex	30	40%	**0.321**	0.303	0.233

The General–SCISOR contrast intervals at 20%, 30%, and 40% removal are $[0.0502, 0.1102]$, $[0.0579, 0.1188]$, and $[0.0786, 0.1339]$. The corresponding Complex–SCISOR intervals are $[-0.0391, 0.0357]$, $[-0.0206, 0.0386]$, and $[-0.0164, 0.0529]$. At the original 20%, 30%, and 40% budgets, all six General-60 and Complex-30 TM contrasts against Raygun are positive with intervals excluding zero. WT contact recovery is also challenging under the complex predictor: it averages 13.10% on the screened complex panel, with zero recovery in 11 of 30 cases, and 28.79% on 149 evaluable primary complexes, with zero recovery in 51 cases.

Table 12: **Native interface-contact recovery on the family-screened Complex-30 cohort.** †: identification-bound midpoint.

Removed	SHRINKFLOW ↑	SCISOR ↑	Raygun-500 ↑
20%	4.66	**4.79**	2.71†
30%	**6.28**	2.61	1.83†
40%	**6.17**	1.33	1.50†

D.3 RAYGUN SHORT-INPUT REPAIR

Native Raygun-500 fails in two General-60 conditions at 60% removal and ten at 70%. The failure occurs when iterative re-encoding receives sequences of length 36–48, below the implementation's fixed 50-bin pooling configuration, producing zero-width windows. For inputs shorter than 50 residues, the repaired implementation uses identity reduction; inputs of length at least 50 follow the original path unchanged.

The checkpoint, random seed, 500-candidate budget, batch size, noise setting, decoding rule, number of iterations, and native PLL-based selection remain unchanged. Boundary tests cover 14 sequence lengths, and the unchanged path is elementwise identical to the native implementation.

At 60% and 70% removal, the native implementation produces evaluable outputs for 58 and 50 of 60 proteins, with mean full-WT TM 0.17693 and 0.14479. The repaired full-cohort results are 0.17844 and 0.14859, respectively. Native failures are retained as non-deliveries rather than assigned zero structural quality.

D.4 FUNCTIONAL-SITE RETENTION WITHOUT PROTECTION

We evaluate two independently curated 20-protein panels at 20% and 40% removal. Both methods produce eight native candidates per protein and deletion budget. A shared selector uses the raw arithmetic mean of the three PLM axes to choose one candidate, without using functional annotations during generation or selection. Each panel therefore contains 640 generated candidates and 80 selected outputs across the two methods and two budgets.

M-CSA panel. Catalytic annotations are drawn from a mechanism/catalytic-residue snapshot dated 2026-08-26, with UniProt sequences and PDB-coordinate correspondence. After homology screening against training data and earlier evaluation panels, followed by family deduplication, 20 proteins are selected in a fixed order from the remaining eligible set.

UniProt panel. The panel is curated from UniProtKB reviewed release 2026_03. The source query retrieves the first 500 accession-sorted enzyme records satisfying length 100–800 and experimental active-site and binding-site criteria. Active-site positions require `ECO:0000269`; binding positions require `ECO:0000269` or PDB-supported `ECO:0007744`. Excluding M-CSA reference IDs leaves 163 records; homology screening against training data and previous panels leaves 68 records forming 63 internal clusters. Fixed hashing and length stratification select 20 proteins. The resulting set is a deterministic curated panel rather than a population-representative sample of UniProt enzymes. Thirteen proteins contain only one experimentally supported active-site annotation, so the endpoint measures retention of recorded annotations rather than completeness of the catalytic center.

The released SHRINKFLOW checkpoint uses rate temperature three, 32 integration steps, and minimum interval length three. SCISOR uses its original U90S model, native position-wise deletion, and temperature one. Table 13 reports selected-output retention.

Table 13: **Functional-site retention without protection.** Each panel has 20 proteins and uses the same best-of-eight raw-PLM selector for both methods. Values are percentages; annotations are used only for evaluation.

Panel / endpoint	Removed	SHRINKFLOW ↑	SCISOR ↑
M-CSA catalytic residues	20%	78.71	**90.83**
	40%	51.21	**69.29**
UniProt catalytic/binding union	20%	85.11	**90.18**
	40%	58.52	**62.86**
UniProt active-site positions	20%	85.00	**98.33**
	40%	**66.67**	60.83

Averaging the two budgets within each protein, SHRINKFLOW retains fewer M-CSA catalytic residues than SCISOR by 15.10 percentage points (95% paired protein interval $[-25.21, -4.17]$). The corresponding difference on the UniProt catalytic/binding union is smaller at -4.71 points (95% interval $[-14.51, 4.01]$). These results reinforce the distinction between global structural preservation and retention of sparse annotated functional positions.

D.5 LOCAL STRUCTURE UNDER SHARED SITE-PROTECTION CONSTRAINTS

We next evaluate a constrained setting in which both methods receive the same catalytic positions and specified cofactor/metal-contact positions as non-deletable residues. The experiment contains 20 enzymes evaluated at 20% and 40% removal. Each method generates eight candidates per protein and budget, and the same frozen PLM scorer selects one candidate. This yields 640 generated candidates and 80 selected shortened sequences; 20 WT structures are additionally predicted for reference. All structures are predicted with the same single-chain Boltz-2 configuration using empty MSA, three recycles, 200 sampling steps, and one structural sample.

The prespecified primary endpoint is a fixed-denominator local Cα score around the catalytic region. The neighborhood contains residues whose Cα atoms lie within 8 Å of a catalytic Cα atom in the experimental structure. Reference residue pairs within 15 Å are evaluated at distance-error

thresholds of 0.5, 1, 2, and 4 Å. Pairs with a deleted endpoint contribute zero while the reference denominator remains fixed, so the score jointly reflects preservation of local coverage and geometry. We additionally report full-WT-normalized TM and catalytic-geometry MAE using either the experimental WT structure or the predicted WT structure as reference.

Table 14: **Structure around shared protected functional positions.** Both methods receive the same catalytic and specified cofactor/metal-contact positions as non-deletable residues. Lower geometry MAE is better.

Removed	Method	Local Cα score ↑	Full-WT TM ↑	Geometry MAE Exp. WT (Å) ↓	Geometry MAE Pred. WT (Å) ↓
20%	SHRINKFLOW	**0.494**	**0.399**	**3.954**	**2.696**
	SCISOR	0.355	0.345	5.110	4.265
40%	SHRINKFLOW	**0.475**	**0.327**	**4.648**	**3.996**
	SCISOR	0.180	0.251	7.090	6.180
Avg.	SHRINKFLOW	**0.485**	**0.363**	**4.301**	**3.346**
	SCISOR	0.267	0.298	6.100	5.222

Averaging the two removal budgets within each protein, SHRINKFLOW improves the fixed-denominator local Cα score by $+0.2176$ (95% CI $[0.1346, 0.3102]$), with higher scores on 19 of 20 proteins. The corresponding full-WT TM difference is $+0.0653$ ($[0.0213, 0.1094]$). Catalytic-geometry MAE is lower by 1.7991 Å relative to the experimental WT reference ($[-3.0477, -0.5849]$) and by 1.8765 Å relative to the predicted WT reference ($[-3.0910, -0.5909]$).

The effect is particularly pronounced at 40% removal. The local Cα score difference is $+0.2957$ ($[0.1923, 0.4116]$), with SHRINKFLOW higher on 19 of 20 proteins. At 20% removal, the corresponding difference is $+0.1395$ ($[0.0450, 0.2415]$).

Averaged across the two budgets, local residue retention is 92.33% for SHRINKFLOW and 79.54% for SCISOR, while fixed local-pair retention is 88.21% and 67.49%, respectively. Both methods retain all supplied catalytic and protected cofactor/metal-contact positions and satisfy the requested deletion budget in every evaluated condition. The local-context measurements therefore compare how well the two shrinking procedures preserve the structural neighborhood surrounding the same protected functional positions, rather than differences in whether those protected positions themselves remain present.

Additional protected-site controls. We compare first unranked outputs ($K = 1$) from SHRINKFLOW, SCISOR, legal-reference sampling, and morphology-matched random deletion under the same site-protection constraints. The morphology-matched control preserves the deletion morphology of SHRINKFLOW while randomizing deletion placement subject to those constraints.

Table 15: **Protected-site controls without candidate reranking.** The primary endpoint is the fixed-denominator local Cα score.

Method	20% ↑	40% ↑	Avg. ↑
SHRINKFLOW	0.492	**0.443**	**0.467**
SCISOR	0.336	0.168	0.252
Legal reference	0.381	0.279	0.330
Morphology-matched random	**0.494**	0.384	0.439

Averaging the two budgets within each protein, the local Cα score differences are $+0.2156$ against SCISOR (95% CI $[0.1269, 0.3124]$), $+0.1376$ against legal-reference sampling ($[0.0586, 0.2287]$), and $+0.0279$ against morphology-matched random deletion ($[-0.0154, 0.0815]$).

Terminal truncation is feasible for 18 of 20 proteins at 20% removal and 15 of 20 at 40%. Among the 15 proteins feasible at both budgets, its budget-averaged local Cα score is 0.6025, compared with 0.5119 for SHRINKFLOW, corresponding to a difference of $+0.0907$ in favor of terminal truncation (95% CI $[0.0179, 0.1779]$).

D.6 Exploratory Analysis of Site Retention

We first examine whether final candidate selection explains the functional-site retention gap. Before selection, budget-averaged M-CSA catalytic retention is 66.05% for SHRINKFLOW and 75.77% for SCISOR. On UniProt, functional-union retention is 68.70% versus 74.93% before selection and 71.82% versus 76.52% after selection. The difference is therefore already present in the generated candidate pools.

Within SHRINKFLOW candidate pools, the mean Spearman association between PLM score and functional-site retention is modest: approximately 0.165 on M-CSA and 0.116 on UniProt. These correlations indicate that the generic PLM score is only weakly associated with retention of the sparse annotations used in these panels. The score contains no explicit site-specific term. The two methods also differ substantially in deletion patterns, motivating the morphology-matched controls in Appendix D.5.

Across the two panels, SHRINKFLOW produces approximately 2.1–2.4 deletion blocks per selected sequence, with mean longest-block lengths of 106–111 residues. SCISOR produces approximately 73–74 blocks, with mean longest-block lengths near six residues. In the 18 M-CSA protein–budget conditions where SHRINKFLOW removes at least two catalytic positions, those positions are removed within a single deletion block.

E Complete-Outcome Evaluation and Feedback Distillation

E.1 Complete-Outcome Scoring Rules

The fixed-pool scoring study uses identical General-240 candidate pools at 30% removal and changes only the rule used to rank candidates. We compare the three individual scoring axes, leave-one-axis-out variants, the raw arithmetic mean, the calibrated score used by the primary method, and WT-only evaluation.

Table 16: **Complete-outcome scoring rules on General-240.** All rules select from identical candidate pools at 30% removal. Bold indicates the largest displayed mean within each metric.

Rule	Full-WT TM ↑	Cα-lDDT ↑	Contacts (%) ↑
U_g only	0.574	0.741	48.95
U_p only	0.572	0.745	49.12
U_r only	0.579	**0.760**	**50.33**
Full calibrated score	0.576	0.755	49.79
Raw mean	**0.580**	0.759	50.31
Without U_g	0.578	0.757	50.08
Without U_p	0.578	0.756	49.97
Without U_r	0.573	0.744	48.89
WT-only	0.476	0.545	35.99

All tested complete-outcome scoring variants yield higher mean full-WT TM, Cα-lDDT, and native-contact recovery than WT-only scoring. Differences among the individual-axis and combined scores are smaller than their differences from WT-only (Table 16).

E.2 Cross-PLM Robustness of Complete-Outcome Selection

We next test whether the value of complete-outcome candidate evaluation is specific to the ESMC-600M scorer used by the primary method. The SHRINKFLOW generator, candidate pools, and calibration plans are held fixed: for each protein and removal level, the same 32 generated candidates and eight calibration plans are rescored independently with ESMC-600M and ESM-2 650M. Each scorer then selects one candidate from the same pool. This experiment changes only the candidate-selection model; the generator is not retrained with ESM-2 supervision.

Enzyme-120. Table 17 compares the first unranked candidate with ESMC- and ESM-2-selected candidates on 120 enzymes. Complete-site retention measures the fraction of proteins for which all

annotated catalytic positions are retained and is evaluable for 119 proteins; full-WT TM uses all 120 proteins.

Table 17: **Cross-PLM complete-outcome selection on Enzyme-120.** The same 32 SHRINKFLOW candidates are rescored by ESMC-600M and ESM-2 650M. Complete-site retention is evaluated on 119 proteins; full-WT TM uses all 120 proteins.

Removed	Selection	Complete-site retention (%) ↑	Full-WT TM ↑
20%	First candidate	37.82	0.623
	ESMC, $K = 32$	53.78	0.690
	ESM-2, $K = 32$	**54.62**	**0.699**
30%	First candidate	19.33	0.520
	ESMC, $K = 32$	35.29	0.589
	ESM-2, $K = 32$	**36.13**	**0.604**
40%	First candidate	11.76	0.430
	ESMC, $K = 32$	19.33	0.496
	ESM-2, $K = 32$	**20.17**	**0.516**

Both complete-outcome scorers select candidates with higher mean full-WT TM than the first unranked output at all three removal levels. Complete-site retention also increases substantially relative to the first candidate. Direct ESM-2–ESMC differences in complete-site retention are small at all three removal levels ($+0.84$ percentage points), while the corresponding full-WT TM differences are $+0.009$, $+0.015$, and $+0.020$ at 20%, 30%, and 40% removal. The two scorers show moderate within-pool agreement: mean Spearman correlations are 0.631, 0.625, and 0.623, while same-Top-1 rates are 44.17%, 47.50%, and 49.17%, respectively. Complete-outcome selection therefore remains effective with a second PLM despite frequent differences in the selected Top-1 candidate.

E.3 JOINT VERSUS ADDITIVE OUTCOME EVALUATION

We compare Joint scoring with three additive per-span constructions on a fixed 30-protein panel at 20% and 40% removal. Each protein–budget condition contains independently constructed two-interval and three-interval pools of 16 plans. Within each pool, the total deletion amount, interval count, and interval lengths are fixed; only deletion placement and combinations vary.

Additive scoring rules. For a plan $M = \{I_1, \ldots, I_J\}$, let $u_j = U(x, I_j)$ denote the raw arithmetic mean of the three PLM similarities in Equation 1, evaluated after deleting only interval I_j from the WT sequence. For each protein and exact interval length ℓ, a calibration set $\mathcal{C}_\ell$ contains the scores of up to 32 uniformly sampled legal single-interval placements without replacement, enforcing the same protected positions. All legal placements are used when the support contains fewer than 32. Let μ_ℓ and σ_ℓ be their mean and population standard deviation. The three additive scores are

$$
\begin{aligned}
S_{\text{raw}}(M) &= \frac{1}{J}\sum_{j=1}^{J} u_j, \\
S_{\text{length-z}}(M) &= \frac{1}{J}\sum_{j=1}^{J} \frac{u_j - \mu_{|I_j|}}{\max(\sigma_{|I_j|}, 10^{-6})}, \\
S_{\text{rank}}(M) &= \frac{1}{J}\sum_{j=1}^{J} \frac{\sum_{v \in \mathcal{C}_{|I_j|}} \left[\mathbf{1}\{v < u_j\} + \frac{1}{2}\mathbf{1}\{v = u_j\}\right]}{|\mathcal{C}_{|I_j|}|}.
\end{aligned}
\tag{10}
$$

Joint ranks the same candidate pool by $U(x, M)$, the raw three-axis mean for the complete shortened sequence. Each rule selects the largest score, breaking ties by the smallest candidate index. Calibration statistics are specific to single-deletion length; the separately computed joint calibration is not applied to single-interval scores.

Native-contact differences are paired within each protein, budget, and pool type, then averaged across pool types and budgets before protein-level bootstrap resampling. Joint improves selection over WT-only but shows no clear advantage over the three tested additive constructions (Table 18).

Table 18: **Joint versus additive candidate selection.** Entries are Joint minus the indicated scoring rule in fixed-WT-denominator native-contact recovery. Results average pool types and budgets within each protein; $n = 30$. Brackets give 95% protein-bootstrap intervals.

Comparison	Difference [95% CI]
Joint − length-calibrated Additive	-0.002 $[-0.013, +0.008]$
Joint − raw Additive	-0.000 $[-0.011, +0.009]$
Joint − rank Additive	-0.004 $[-0.017, +0.008]$
Joint − WT-only	$+0.044$ $[+0.025, +0.065]$

E.4 Matched Training-Supervision Controls

We evaluate whether complete-outcome feedback changes the learned proposal distribution under the primary proposal-corrected training recipe. Four matched models share the same generator architecture, capability initialization, training proteins, candidate pools, proposal correction μ/q, optimizer, 20,000-update budget, and inference protocol. Full uses complete-outcome plan quality; NoReward removes the quality term while retaining proposal correction; Shuffled permutes complete-outcome quality within each protein–budget pool; and WT-only scores each deletion plan using only WT representations, taking the cosine similarity between the mean embedding over retained WT positions and the mean embedding over the full WT sequence. The WT-only comparison therefore contrasts two supervision constructions rather than isolating re-encoding as a single factor.

Each supervision variant is evaluated across three matched training seeds. For each seed, the model produces eight fresh, unselected outputs per protein and deletion budget. Per-protein metrics average these outputs and then the three training seeds, following Appendix C. The primary rate temperature is $T = 3$, with $T = 1$ used for sensitivity analysis. Results use 58 general proteins and 29 protein complexes after excluding two general proteins and one complex with MMseqs2 matches to the training/validation exposure union (identity $\geq 30\%$, query coverage $\geq 80\%$, $E \leq 0.001$).

Table 19: **Matched supervision controls at the primary $T = 3$ setting.** Entries are Full minus the indicated control for the primary endpoint: fixed-denominator native-contact recovery on General-58 and native interface-contact recovery on Complex-29. Brackets give 95% paired protein-bootstrap intervals.

Panel	Removed	Δ vs. WT-only	Δ vs. NoReward	Δ vs. Shuffled
General	20%	$+0.007$ $[+0.004, +0.010]$	$+0.002$ $[-0.002, +0.006]$	$+0.007$ $[+0.003, +0.011]$
	30%	$+0.006$ $[+0.002, +0.010]$	$+0.004$ $[+0.000, +0.008]$	$+0.002$ $[-0.003, +0.007]$
	40%	$+0.004$ $[+0.000, +0.008]$	$+0.004$ $[+0.000, +0.008]$	$+0.007$ $[+0.004, +0.010]$
Complex	20%	$+0.010$ $[+0.001, +0.018]$	$+0.006$ $[+0.001, +0.011]$	$+0.006$ $[-0.002, +0.014]$
	30%	$+0.002$ $[-0.005, +0.010]$	-0.002 $[-0.008, +0.005]$	-0.002 $[-0.008, +0.003]$
	40%	-0.001 $[-0.006, +0.005]$	-0.001 $[-0.005, +0.003]$	-0.002 $[-0.008, +0.005]$

On General-58, Full improves native-contact recovery over WT-only supervision at all three removal levels, by $+0.0071$, $+0.0061$, and $+0.0040$ at 20%, 30%, and 40% removal, respectively. The contrasts against NoReward and Shuffled are also positive in several budgets but are less uniform (Table 19). Full–WT-only mean differences are also positive for retained-residue lDDT and full-WT TM at all three removal levels; native-contact recovery remains the prespecified primary endpoint.

On Complex-29, the clearest effect occurs at 20% removal, where Full improves native interface-contact recovery over WT-only by $+0.0095$ and over NoReward by $+0.0059$. At 30% and 40% removal, the effects are smaller and mixed across supervision variants.

Temperature sensitivity. At $T = 1$, the Full-minus-WT-only General contact contrasts become larger: $+0.0141$, $+0.0178$, and $+0.0129$ at 20%, 30%, and 40% removal. The corresponding temperature-difference contrasts are positive in all three budgets. The Complex pattern does not strengthen similarly: its $T = 1$ Full-minus-WT-only interface contrasts are -0.0015, -0.0111, and -0.0051. The released model uses $T = 3$ as its primary inference setting; the temperature sensitivity differs substantially between the General and Complex panels.

Orthogonal endpoints. The fixed-native-denominator Cα score on General-58 gives Full-minus-WT-only differences of $+0.0064$, $+0.0068$, and $+0.0037$ across the three removal levels, with paired intervals above zero. ESM-2 single-mask pseudo-log-likelihood also gives positive contrasts in several General and Complex settings. Conservation-weighted retention shows no consistent Full advantage.

Training-weight shift. Across the full frozen training-label pool, the total-variation distances between Full and NoReward candidate weights are 0.293, 0.283, and 0.273 at 20%, 30%, and 40% removal, respectively. The corresponding Full–WT-only distances are 0.416, 0.406, and 0.400. These distances quantify how complete-outcome quality redistributes training mass over the frozen plan pool. The resulting structural changes are smaller because the PLM preference is an indirect training signal for the structural endpoints used at evaluation.

F Candidate Generation Analyses

F.1 Numerical and Budget Controls

On 12 fixed proteins, increasing flow integration from 8 to 32 steps changes candidate-mean Cα-lDDT by $+0.0101$ (95% CI $[-0.0114, 0.0292]$), while the PLM-selected and oracle-best differences are $+0.1340$ ($[0.0603, 0.2308]$) and $+0.0334$ ($[0.0123, 0.0568]$), respectively. The primary model therefore uses 32 integration steps.

We also generate 32 plans for each of 240 proteins at removal ratios from 10% to 80%, totaling 61,440 deletion masks. All generated masks satisfy the requested integer deletion budget and legal interval constraints.

F.2 Shared-Scorer Proposal Analysis

The shared-scorer comparison uses 12 general proteins and 12 enzyme sources from a 48-protein development panel. For the Flow-versus-reference study, each strategy produces 32 plans per protein and deletion budget, with eight independently sampled calibration plans shared within each condition. The exact reference first samples an interval count uniformly and then samples a legal deletion plan uniformly conditional on that count.

We additionally include PLM-guided local search in the main strategy comparison. PLM-local search begins from 512 WT-only mask evaluations computed from a single WT representation, then uses its complete-outcome query budget for the starting plan and subsequent local revisions. These initial WT-only mask evaluations do not require re-encoding shortened sequences.

All strategies use the same calibrated three-axis complete-outcome scorer for final selection. We compare selected structural quality across candidate budgets using exact deletion-mask correspondence and three-recycle ESMFold.

At $K = 32$, Flow-minus-reference TM differences are $+0.031$, $+0.057$, and $+0.072$ at 20%, 30%, and 40% removal, with 95% intervals $[-0.006, 0.068]$, $[0.027, 0.093]$, and $[0.033, 0.111]$, respectively. The separation from exact reference sampling is larger at the higher removal levels.

The first Flow output uses one WT encoding without complete-outcome scoring of edited candidates. Compared with exact reference sampling followed by $K = 32$ selection, its mean TM differences are -0.00576, -0.02326, and -0.00802 at 20%, 30%, and 40% removal, respectively. These differences are small relative to the gains observed after shared candidate selection.

F.3 Terminal-Truncation Comparisons

Terminal truncation allocates the entire deletion budget between the N- and C-termini without removing internal intervals. We compare this strategy with ShrinkFlow on General-240 and Complex-150.

General-240. We compare ShrinkFlow with terminal truncation under a strict matched-candidate protocol on the 240 single-domain General proteins at 20%, 30%, and 40% removal,

Table 20: **Shared-selector candidate quality on 24 proteins.** μ denotes exact reference sampling. K is the candidate count, TM is normalized by full WT length, and ΔTM is Flow minus reference. Bold indicates the higher displayed TM within each row.

Removed	K	μ TM ↑	Flow TM ↑	ΔTM ↑
20%	1	0.474	**0.582**	0.108
	4	0.533	**0.597**	0.064
	8	0.543	**0.597**	0.055
	32	0.588	**0.619**	0.031
30%	1	0.394	**0.475**	0.081
	4	0.431	**0.524**	0.092
	8	0.454	**0.535**	0.081
	32	0.498	**0.555**	0.057
40%	1	0.280	**0.381**	0.101
	4	0.327	**0.424**	0.097
	8	0.347	**0.410**	0.064
	32	0.389	**0.461**	0.072

with $K \in \{1, 8, 32\}$. Terminal candidates are placed in a fixed random order without WT-based prescreening; when fewer than K legal terminal plans exist, all distinct plans are used. The same complete-outcome scorer is used for candidate selection at $K = 8$ and $K = 32$, while $K = 1$ reports the first unranked candidate.

Across all nine budget–K conditions, terminal truncation achieves higher native-contact recovery, retained-residue Cα-lDDT, and full-WT-normalized TM than SHRINKFLOW. Table 21 summarizes the paired native-contact differences. The corresponding TM and lDDT comparisons follow the same direction, with paired confidence intervals excluding zero in favor of terminal truncation.

Table 21: **General-240 terminal-truncation comparison.** Entries are paired differences in native-contact recovery, SHRINKFLOW minus terminal truncation. Negative values favor terminal truncation.

Removed	K	Δ Contacts [95% CI]
20%	1	-0.136 $[-0.154, -0.118]$
	8	-0.084 $[-0.098, -0.070]$
	32	-0.057 $[-0.068, -0.046]$
30%	1	-0.139 $[-0.157, -0.121]$
	8	-0.080 $[-0.096, -0.065]$
	32	-0.051 $[-0.064, -0.038]$
40%	1	-0.112 $[-0.128, -0.096]$
	8	-0.076 $[-0.088, -0.063]$
	32	-0.041 $[-0.051, -0.030]$

General-240 consists of single CATH-domain inputs and imposes no requirement for internal or distributed deletions.

Complex-150. On Complex-150, Flow–Terminal differences in native interface-contact recovery are statistically unresolved across all nine budget–K conditions under the native terminal-candidate protocol.

Scope under shared protection constraints. On a separate 36-protein panel, unconstrained terminal truncation admits a legal solution for all proteins across 20–70% removal. With shared protected positions enforced, the corresponding feasible counts are 28, 23, 20, 17, 13, and 12 out of 36 proteins. SHRINKFLOW and SCISOR satisfy the same protection constraints and exact deletion budgets in all 36 proteins at every removal level. Allowing terminal deletions of length one rather than three leaves these feasibility counts unchanged.

Terminal-infeasible requests are reported as missing rather than assigned zero structural quality. On the terminal-feasible subsets, terminal truncation remains structurally strong. For example, at 50% removal its mean local Cα score is 0.662 on the 17 feasible proteins, compared with 0.520 for

SHRINKFLOW and 0.095 for SCISOR. Together, these comparisons show two complementary properties of terminal-only deletion: it is structurally strong when feasible, while its feasible coverage decreases as protection constraints and deletion budgets become more restrictive.

F.4 STRONG INTERNAL-DELETION BASELINES

We compare internal-deletion strategies on 36 protected proteins at 20%, 30%, and 40% removal, averaging budgets within each protein before paired bootstrap resampling. First-output comparisons show higher native-contact recovery for SHRINKFLOW than SCISOR and the protected reference μ. Differences from the stronger simple strategies depend on the endpoint: terminal-prior greedy deletion has higher native-contact recovery, while the local Cα difference is small. Under common $K = 32$ selection, the native-contact difference between SHRINKFLOW and sequential-uniform legal sampling is also small (Table 22).

Table 22: **Protected internal-deletion comparisons.** Entries are SHRINKFLOW minus the indicated baseline, averaged across 20%, 30%, and 40% removal within each protein. $n = 36$; brackets give 95% paired protein-bootstrap intervals.

Protocol	Baseline	Local Cα difference	Contact difference
First output	SCISOR	$+0.299$ $[+0.239, +0.364]$	$+0.164$ $[+0.137, +0.191]$
	Protected μ	$+0.243$ $[+0.184, +0.308]$	$+0.167$ $[+0.138, +0.197]$
	Morphology-matched	$+0.008$ $[-0.032, +0.050]$	$+0.021$ $[+0.004, +0.037]$
	Sequential-uniform	$+0.033$ $[-0.011, +0.079]$	$+0.012$ $[-0.012, +0.036]$
	Terminal-prior greedy	$+0.043$ $[-0.050, +0.145]$	-0.091 $[-0.115, -0.066]$
Shared $K = 32$	Sequential-uniform	$+0.020$ $[-0.017, +0.056]$	-0.009 $[-0.027, +0.008]$

F.5 CAPABILITY-STAGE REFERENCE COMPARISON

On the fixed 30-protein panel, the released policy and the capability-stage reference both use minimum interval length three. For first unselected outputs, the released-policy minus reference native-contact difference is $+0.03236$ (95% CI $[0.00674, 0.06097]$). After common best-of-eight selection, the difference is $+0.01810$ ($[-0.00708, 0.04272]$).

This comparison is used as a stage-level reference for the released policy. Component-specific effects are evaluated separately by the matched supervision controls in Appendix E.

G AF3 EVALUATION ON RAYGUN CASE-STUDY TARGETS

We evaluate eight targets appearing in Raygun case studies: HBB, CCR1, lacZ, mTOR, eGFP, mCherry, TurboID, and SpCas9 (Devkota et al., 2026). Within each target, SHRINKFLOW, SCISOR, and Raygun use the same WT sequence and target length and are evaluated under the same AlphaFold 3 refolding protocol. Each reported value is the mean over eight evaluated candidates per method. HBB, CCR1, lacZ, mTOR, eGFP, mCherry, and TurboID are evaluated as single-chain proteins, whereas SpCas9 is evaluated together with its DNA and RNA partners as a ternary complex.

Refolding and visualization. Each WT or shortened-sequence prediction uses one model seed and one diffusion sample per seed. SpCas9 inputs contain protein chain A, DNA strands B and C, and sgRNA chain D. The nucleic-acid partner sequences are identical across the WT and shortened designs, and WT references use the same prediction configuration as the corresponding shortened sequences. Figure 3 displays candidate `00`, the first returned candidate from each method. Metrics shown in the figure refer to those individual structures; aggregate results average the eight evaluated candidates per method.

For each candidate, AF3 reports pLDDT, pTM, and ranking score. Structural preservation is evaluated relative to the corresponding same-protocol AF3 WT prediction using full-WT-normalized TM, retained-residue Cα-lDDT, and native-contact recovery. Native-contact recovery uses a fixed WT denominator, so contacts involving deleted or unmapped residues count as unrecovered.

Table 23: **AF3 evaluation on Raygun case-study targets.** Values are means over eight evaluated candidates per method. pLDDT, pTM, and Rank are AF3 confidence measures; TM, Cα-lDDT, and Contacts are computed relative to the corresponding same-protocol AF3 WT prediction.

Case (removed)	Method	pLDDT ↑	pTM ↑	Rank ↑	TM ↑	Cα-lDDT ↑	Contacts ↑
HBB (15.7%)	SHRINKFLOW	**66.37**	**0.523**	**0.533**	**0.725**	**0.784**	**0.656**
	SCISOR	53.99	0.378	0.383	0.600	0.570	0.498
	Raygun	57.84	0.444	0.471	0.643	0.651	0.512
CCR1 (16.9%)	SHRINKFLOW	69.13	0.664	0.759	0.701	**0.863**	**0.678**
	SCISOR	**70.76**	**0.731**	**0.795**	**0.704**	0.743	0.618
	Raygun	45.47	0.311	0.430	0.421	0.404	0.424
lacZ (14.7%)	SHRINKFLOW	**88.64**	**0.908**	**0.918**	**0.841**	**0.952**	**0.735**
	SCISOR	74.52	0.833	0.835	0.801	0.746	0.585
	Raygun	28.18	0.164	0.531	0.194	0.146	0.044
mTOR (24.8%)	SHRINKFLOW	**78.46**	**0.704**	**0.748**	**0.664**	**0.956**	**0.704**
	SCISOR	50.50	0.319	0.374	0.360	0.487	0.372
	Raygun	51.25	0.194	0.611	0.112	0.737	0.005
eGFP (10.9%)	SHRINKFLOW	**80.58**	**0.805**	**0.821**	**0.811**	**0.854**	**0.695**
	SCISOR	70.03	0.711	0.711	0.783	0.684	0.606
	Raygun	59.50	0.614	0.625	0.720	0.583	0.539
mCherry (15.7%)	SHRINKFLOW	**81.87**	**0.812**	**0.820**	**0.795**	**0.906**	**0.679**
	SCISOR	61.82	0.605	0.634	0.693	0.593	0.497
	Raygun	43.82	0.451	0.555	0.546	0.424	0.345
TurboID (5.9%)	SHRINKFLOW	**84.09**	**0.856**	**0.863**	**0.907**	**0.935**	**0.836**
	SCISOR	83.64	0.837	0.837	0.881	0.843	0.788
	Raygun	71.61	0.751	0.785	0.807	0.784	0.720
SpCas9 (28.9%)	SHRINKFLOW	**77.71**	**0.708**	**0.794**	**0.596**	**0.912**	**0.601**
	SCISOR	45.06	0.455	0.630	0.387	0.411	0.277
	Raygun	32.84	0.205	0.405	0.186	0.249	0.117

For the SpCas9–DNA–sgRNA complex, mean ipTM is 0.806 for SHRINKFLOW, 0.671 for SCISOR, and 0.186 for Raygun. Across the eight evaluated targets, SHRINKFLOW achieves the highest mean pLDDT and full-WT TM on seven, and the highest mean retained-residue Cα-lDDT and native-contact recovery on all eight.

H COMPUTATIONAL RESOURCES AND REPRODUCIBILITY

Primary quality-supervised training uses 20,000 optimizer updates on four A100 GPUs. Main sequence generation is performed on A100 GPUs; structural evaluations use A40 or A100 GPUs according to the predictor configuration specified in the corresponding experiment. Matched training-supervision controls follow the training and inference protocol in Appendix E.

The supplementary material provides the SHRINKFLOW implementation, paper-scale configurations, validation tests, and reproduction instructions. Full structural benchmark reproduction additionally requires the external datasets, model checkpoints, and third-party prediction resources described above.