STEER, SELECT, AND STRATIFY: TRAINING-FREE SAMPLING FOR PROTEIN MOTIF SCAFFOLDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Motif scaffolding embeds functional sites with specified geometry into protein scaffolds for therapeutics and biocatalysis. RFdiffusion is widely used for this task. However, two challenges remain: obtaining successful designs on difficult tasks and covering diverse successful structures within a finite candidate budget. We further diagnose the first challenge into two bottlenecks: generated-motif errors are associated with design failure, while accurate motif geometry alone does not ensure structural recovery after sequence design and refolding. For the second challenge, independent sampling can repeatedly produce similar successful structures, limiting finite-batch coverage. We therefore propose S^3-Scaffold (Steer, Select, and Stratify), a training-free sampling framework with three corresponding components: gated gradient guidance, rollout reweighting, and Exploration via Marginal-Preserving Stratification (EMPS). Gated gradient guidance restores truncated geometric gradients without changing the denoiser's forward computation and uses clean predictions to guide scaffold updates gated by denoising time and motif error. Rollout reweighting combines scaffold consistency and refolded-motif errors from sequence design and refolding into a parent score for allocating independent continuations. EMPS jointly stratifies scaffold layouts and a geometry-dependent process coordinate of translational noise without additional model evaluations. Before rollout reweighting, we prove that EMPS preserves matched-reference marginals and expected success count while providing expected coverage of fixed successful categories no worse than independent or single-axis sampling. For the full framework, we establish finite-batch lower bounds on expected coverage under anchor-retention and successful-continuation conditions. Across all 30 MotifBench tasks at the standard budget of 100 backbones per task, S^3-Scaffold raises RFdiffusion's published score from 21.95 to 30.27, solved tasks from 17 to 22, and successful clusters from 205 to 225. Component ablations and mechanism diagnostics support the complementary roles of geometric control, downstream feedback, and coordinated exploration.

1 INTRODUCTION

De novo protein design creates opportunities for therapeutics and biocatalysis (Cao et al., 2020; Lauko et al., 2025). Many applications require embedding a functional motif with specified geometry into a supporting scaffold, a problem known as motif scaffolding (Wang et al., 2022). RFdiffusion has substantially advanced backbone generation for this task (Watson et al., 2023). Yet evaluations of the original RFdiffusion show that many sampled backbones fail end-to-end sequence-design and refolding criteria, while successful outputs can concentrate within a limited set of scaffold structures (Watson et al., 2023; Yim et al., 2024; Zheng et al., 2025). These limitations motivate improving the sampling process itself. Training-free sampling keeps pretrained model weights fixed while incorporating task-specific objectives or feedback during inference, thereby allowing the same generator to be reused across design objectives (Bansal et al., 2023; Wu et al., 2023; Kim et al., 2025). In our setting, this allows geometric and downstream design feedback to guide RFdiffusion without retraining it. We study this approach with RFdiffusion, aiming to increase both design success and coverage of distinct successful structures for downstream screening at a fixed candidate count.

Two high-level challenges motivate our design. The first is obtaining successful designs on difficult motif-scaffolding tasks. We decompose this challenge into two bottlenecks. **Motif geometry**: fixing

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.

the target motif in RFdiffusion's input does not ensure that the denoiser's clean prediction preserves its geometry. **Refolding compatibility**: accurate motif geometry in a generated backbone does not ensure that a designed sequence will refold to recover both the scaffold and the original target motif. The second challenge is **successful-structure diversity**: independent sampling can repeatedly produce similar successful scaffolds, limiting the number of distinct successful structures obtained from a fixed candidate budget. Accordingly, geometric control and downstream feedback target the two success bottlenecks, while coordinated exploration targets successful-structure diversity.

We propose S^3-Scaffold, a training-free RFdiffusion sampling framework comprising three complementary, independently usable components. **Gated gradient guidance** uses clean predictions as differentiable feedback to translate and rotate free scaffold residues, keeping the input motif fixed. We restore geometric derivatives truncated by the released RFdiffusion implementation without changing its forward values. Denoising time and recorded motif error gate the displacement budget. **Rollout reweighting** evaluates intermediate clean structures through sequence design and refolding. It combines per-sequence scaffold consistency and refolded-motif errors into a parent score, then allocates independent continuations from saved diffusion states. **Exploration via Marginal-Preserving Stratification (EMPS)** jointly stratifies scaffold layouts and a geometry-dependent process coordinate of translational noise. It samples one independent conditional trajectory per equal-probability joint stratum and uniformly permutes them across batch slots, without additional model evaluations.

Our theory establishes sufficient conditions for local motif-error reduction under guidance and relates conditional expected success gains from reweighting to score–success covariance. Before reweighting, EMPS preserves the single-trajectory distribution and the batch's expected success count. Its expected coverage of fixed successful categories is not lower than independent sampling or either single-axis stratification. Strict gains arise when added stratification distinguishes categories that the comparison scheme can still miss. For S^3-Scaffold, an exact decomposition separates coverage losses from omitted parents and gains from additional independent continuations. Anchor-retention and successful-continuation conditions yield finite-batch lower bounds on expected coverage (Appendix F).

Across all 30 MotifBench tasks at the standard budget of 100 backbones per task (Zheng et al., 2025), S^3-Scaffold improves RFdiffusion's published score from 21.95 to 30.27, solved tasks from 17 to 22, and successful clusters from 205 to 225. EMPS alone reaches 23.50 with the same observed successful-design count and improves the exact expected number of successful clusters represented in 50-of-100 subsets of the evaluated pools. Ablations assess endpoint performance; gradient checks, local interventions, downstream failure analyses, and coverage analyses examine the corresponding mechanisms. Our contributions are threefold:

- **Faithful geometric gradients for RFdiffusion.** We restore continuous geometric derivatives without changing forward values and validate translation and rotation gradients through independent finite differences.
- **Diagnosis-driven sampling for successful design.** We characterize failure patterns in RFdiffusion and translate these findings into complementary geometric control and rollout feedback, improving MotifBench performance.
- **Successful-structure diversity through joint stratification.** We introduce EMPS with matched-reference coverage guarantees, derive finite-batch bounds for S^3-Scaffold, and validate successful-structure coverage gains.

2 Related work

Motif scaffolding with pretrained generators and its evaluation. RFdiffusion established conditional backbone generation for motif scaffolding (Watson et al., 2023); subsequent generators extend its conditioning, architecture and scale (Lin et al., 2024; Yim et al., 2024; Geffner et al., 2025; 2026; Zhang et al., 2025; Lu et al., 2025; Lin et al., 2026; Butcher et al., 2025). RFdiffusion2 scaffolds atom-level enzyme active sites without prespecifying residue order (Ahern et al., 2026). MotifBench evaluates ProteinMPNN–ESMFold refolds using geometric criteria and structural clustering (Dauparas et al., 2022; Lin et al., 2023; Zheng et al., 2025); Scaffold-Lab separates designability, novelty, diversity and efficiency (Zheng et al., 2026).

Inference-time geometric control of a fixed generator. Classifier and reconstruction guidance differentiate a noisy-state predictor or a denoised estimate (Dhariwal & Nichol, 2021; Ho et al., 2022), and inverse-problem variants build approximate differentiable objectives (Song et al., 2023). For proteins, conditional diffusion models derive coordinate or $\mathrm{SE}(3)$ corrections from motif-specific objectives (Didi et al., 2023). Other methods act on the denoised output rather than its gradient. Proximal feasibility updates enforce hard structural constraints through ADMM (Christopher et al., 2026), salad aligns and replaces motif coordinates by editing (Jendrusch & Korbel, 2025), and ProHiFlo steers a separately trained hierarchical flow model with pretrained functional predictors (Wang et al., 2026). We instead restore geometric derivatives truncated inside RFdiffusion and move free-residue controls through its Jacobian, leaving its forward values and pretrained weights unchanged. A gate on denoising time and recorded motif error then limits the resulting displacement.

Downstream feedback for search and batch allocation. Feynman–Kac systems formalize sequential proposal, weighting and resampling (Del Moral, 2004), and diffusion variants apply them to protein inpainting and conditional sampling (Trippe et al., 2023; Wu et al., 2023). Young & Akyildiz (2025) compare SMC-aided posterior samplers for zero-shot scaffolding under motif guidance potentials, where the weights target structural conditioning. Closest to our setting, Hartman et al. (2025) steer RFdiffusion particles using sequence-design and relaxation rewards. A different route changes training instead. JointDiff couples residue types, positions and orientations in one diffusion process (Zhu et al., 2025), and ReaPro-1c aligns a generator's intermediate representations with pretrained protein-understanding encoders (Xu et al., 2026). Our rollout reweighting keeps the backbone, sequence-design and refolding models fixed, and uses their feedback only to allocate a finite candidate budget, without claiming exact posterior sampling.

Batch diversity and stratified sampling. Classical stratification spreads samples over input coordinates or Gaussian-path directions (McKay et al., 1979; Glasserman et al., 1999), and Restart perturbs individual trajectories (Xu et al., 2023). Batch-level methods use repulsion or explicit coverage objectives (Corso et al., 2024; Morshed & Boddeti, 2025), and EDDY preserves exact marginals through divergence-free interactions, although its practical approximation relaxes that guarantee (Vinograd et al., 2026). EMPS jointly stratifies scaffold layout and a geometry-dependent process coordinate, preserves the single-trajectory distribution before rollout reweighting, and targets finite-batch coverage of successful structural categories.

3 METHOD

3.1 MOTIF-SCAFFOLDING PROBLEM

Given a target motif, we generate protein backbones that support its geometry and prescribed amino acids. The task specifies motif fragments and allowed lengths for the free segments connecting or flanking them. A layout Q selects these lengths; let L be the resulting residue count, $\mathcal{M} \subseteq \{1, \dots, L\}$ the motif positions, and M the batch size.

Evaluation. For sequence j of backbone i, the scaffold error d_{ij}^{sc} is the refold's mean Cα distance to the generated backbone after full-chain alignment, and the refolded-motif error d_{ij}^{mot} is the refolded motif (N, Cα, C, O)-RMSD to the original target, with all fragments aligned together. A backbone *succeeds* when at least one evaluated sequence satisfies both $d_{ij}^{\mathrm{sc}} < 2\,\text{Å}$ and $d_{ij}^{\mathrm{mot}} < 1\,\text{Å}$, the MotifBench thresholds (Zheng et al., 2025). For a structural grouping, let N_c count successful designs in category $c > 0$. The theory fixes this grouping before sampling, whereas experiments use MotifBench's full-pool clustering. Define the successful-design count and successful-category coverage as

$$S_+ = \sum_{c>0} N_c, \qquad K = \sum_{c>0} \mathbf{1}\{N_c > 0\}. \tag{1}$$

3.2 FROZEN RFDIFFUSION AND SAMPLING STATES

We use the original RFdiffusion as the base model with all parameters frozen; its reverse process updates residue translations and orientations (Watson et al., 2023). We index the T denoising steps

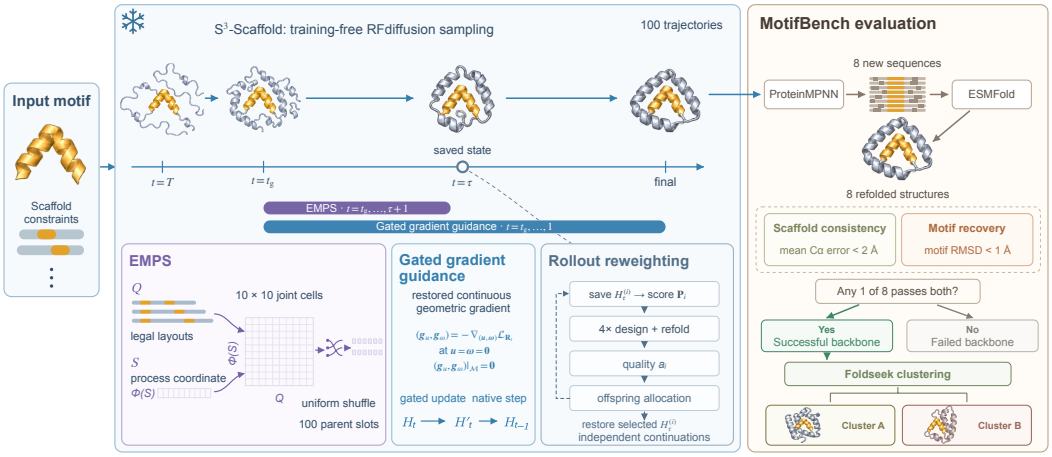

Figure 1: S^3-Scaffold and MotifBench evaluation. Guidance maps H_t to H'_t before RFdiffusion evaluates $D_\theta(H'_t)$. The processed clean coordinates set the EMPS direction when active and form the returned backbone at $t = 1$.

by $t = T, \ldots, 1$. Let $H_t^{(i)}$ be candidate i's complete pre-step state, including every variable read by later transitions; we omit i for one trajectory. The denoiser $D_\theta(H_t)$ returns a raw prediction of the clean backbone: a $3L \times 3$ matrix stacking each residue's N, Cα and C atoms. The native sampler processes this prediction before the next transition, while $t = 1$ returns the clean structure directly. We distinguish the raw prediction used for guidance from the processed clean structure used for rollout scoring and the EMPS direction. Appendix A.1 specifies the state and transition in full.

3.3 S^3-Scaffold Overview

Let t_g be the guidance start and τ the rollout boundary, with $1 \le \tau < t_g \le T$. Figure 1 summarizes the sampling and evaluation workflow. EMPS assigns layout and process-noise strata before initialization and modifies translational noises for $t = t_g, \ldots, \tau + 1$; guidance acts for $t = t_g, \ldots, 1$; and at τ, rollout reweighting scores saved states through sequence design and refolding and allocates M fresh continuations. Appendix A.1 gives the full execution order.

3.4 Gated Gradient Guidance

Although prescribed in the input, motif atoms can deviate in the raw clean prediction. Following reconstruction guidance (Ho et al., 2022), we adjust free residues to reduce this deviation (Figure 2(a)).

Objective and local controls. Let $m = 3|\mathcal{M}|$ count the motif N, Cα, and C atoms, selected by $\mathcal{S}_\mathcal{M} \in \mathbb{R}^{m \times 3L}$. Then $\mathbf{Z}(H) = \mathcal{S}_\mathcal{M} D_\theta(H) \in \mathbb{R}^{m \times 3}$ contains their predicted coordinates and $\mathbf{Y}$ their target coordinates. With centering matrix $\mathbf{C}_m = \mathbf{I}_m - \mathbf{1}\mathbf{1}^\top/m$ and $\varepsilon > 0$, define

$$\mathcal{L}_\mathbf{R}(H) = \left[\frac{1}{m}\|\mathbf{C}_m\mathbf{Z}(H)\mathbf{R} - \mathbf{C}_m\mathbf{Y}\|_F^2 + \varepsilon\right]^{1/2}, \qquad \mathcal{L}(H) = \min_{\mathbf{R}\in SO(3)} \mathcal{L}_\mathbf{R}(H). \tag{2}$$

The control map $\mathcal{T}_{\boldsymbol{u},\boldsymbol{\omega}}$ translates each residue and rotates it about its Cα by axis–angle, and $\Pi_{\mathcal{M}^c}$ zeros motif controls. Holding an optimal alignment $\mathbf{R}_t$ at H_t fixed gives

$$(\boldsymbol{g}_u, \boldsymbol{g}_\omega) = -\Pi_{\mathcal{M}^c} \nabla_{\boldsymbol{u},\boldsymbol{\omega}} \mathcal{L}_{\mathbf{R}_t}(\mathcal{T}_{\boldsymbol{u},\boldsymbol{\omega}} H_t)|_{\boldsymbol{u}=\boldsymbol{\omega}=\mathbf{0}}. \tag{3}$$

Faithful geometric derivatives. Direct differentiation misses coordinate dependencies truncated by the released implementation. For original value y_{ref} and its restored-dependence counterpart y_{geo}, we preserve the former while recovering the latter's derivative through

$$y = \text{sg}(y_{\text{ref}}) + \left[y_{\text{geo}} - \text{sg}(y_{\text{geo}})\right], \tag{4}$$

where sg stops gradients. Appendix B.2 gives the layerwise construction.

Error gate and budget-limited control. Because ungated correction can reduce scaffold consistency, we gate the displacement budget as $D_t = \bar{D}_t \gamma_t(G_t)$, where G_t is the latest recorded motif loss $\mathcal{L}$. With error-gate onset $t_{\text{gate}} = 16$, the logistic gate equals one for $t > t_{\text{gate}}$ and decreases with small G_t thereafter. Normalizing translation and rotation by induced atom displacements ν_u, ν_ω, we apply $H'_t = \mathcal{T}_{\alpha_t \boldsymbol{g}_u/\nu_u, \alpha_t \boldsymbol{g}_\omega/\nu_\omega} H_t$ using a common feasible scale α_t for the displacement and rotation budgets (Appendix B.4).

Remark 1 At $H = H_t$, hold the alignment, gradients, and normalization scales fixed, and define $f_H(a) = \mathcal{L}_{\mathbf{R}_t}(\mathcal{T}_{a\boldsymbol{g}_u/\nu_u, a\boldsymbol{g}_\omega/\nu_\omega} H)$ and $\beta_H = \|\boldsymbol{g}_u\|_2^2/\nu_u + \|\boldsymbol{g}_\omega\|_2^2/\nu_\omega$. Assume exact restored derivatives and unchanged discrete branches along this curve on $[0, \alpha_t]$. If f'_H is absolutely continuous there and $f''_H(a) \le L_H$ almost everywhere for some $L_H \ge 0$, then

$$\mathcal{L}(H'_t) - \mathcal{L}(H_t) \le -\alpha_t \beta_H + \tfrac{1}{2} L_H \alpha_t^2, \tag{5}$$

Strict descent holds for any feasible α_t satisfying $\alpha_t \beta_H > \frac{1}{2} L_H \alpha_t^2$. Appendix B.5 proves the bound using a fixed-alignment upper bound tight at H_t.

3.5 Rollout reweighting

Because low generated-motif error need not ensure downstream success, at τ we save each $H_\tau^{(i)}$, execute the step to obtain $\mathbf{P}_i$, and design and refold four fresh sequences. Using the same-sequence errors from Section 3.1, define

$$r_{ij} = \max\left\{ \frac{d_{ij}^{\text{sc}}}{2\,\text{Å}}, \frac{d_{ij}^{\text{mot}}}{1\,\text{Å}} \right\}, \qquad a_i = \frac{1}{4} \sum_{j=1}^{4} e^{-r_{ij}}. \tag{6}$$

This jointly scores both endpoint criteria. We mix score-proportional and uniform allocation:

$$\pi_i = \frac{\lambda}{M} + (1 - \lambda) \frac{a_i}{\sum_k a_k}, \qquad \lambda = 0.25. \tag{7}$$

Write the parent pool as $\mathcal{H} = (H_\tau^{(i)})_{i=1}^M$ and its scores as $\boldsymbol{a} = (a_i)_{i=1}^M$. Systematic resampling assigns offspring counts $\boldsymbol{n} = (n_i)_{i=1}^M$ with $\sum_i n_i = M$ and $\mathbb{E}[n_i \mid \mathcal{H}, \boldsymbol{a}] = M\pi_i$. Each offspring restores its parent's saved state and executes steps $\tau, \dots, 1$ with fresh randomness. Its eight final evaluation sequences are independent of the four scoring sequences and allocation.

Remark 2 Let $q_{i,+}$ be the unknown success probability of a fresh continuation from parent i, and set $\boldsymbol{q}_+ = (q_{i,+})_{i=1}^M$, $\bar{q}_+ = M^{-1} \sum_i q_{i,+}$, and $\bar{a} = M^{-1} \sum_i a_i$. Relative to assigning one continuation to every parent,

$$\frac{\mathbb{E}[S_+ \mid \mathcal{H}, \boldsymbol{a}]}{M} - \bar{q}_+ = \frac{1 - \lambda}{\bar{a}} \operatorname{Cov}_{\text{emp}}(\boldsymbol{a}, \boldsymbol{q}_+). \tag{8}$$

With the $1/M$ covariance normalization, positive score–success covariance increases conditional expected success (Appendix C.4).

3.6 Exploration via Marginal-Preserving Stratification

EMPS coordinates a batch across layout and process-noise strata while preserving each randomly permuted slot's matched-reference marginal. The reference sampler has the same layout prior, guidance, one continuation per parent, and evaluation, but no rollout reweighting. We denote its path law by $\mathbb{P}_{\text{ref}}$.

Batch allocation. We split the discrete layout prior and a standard-Gaussian process coordinate S into ten equal-probability strata each, forming 100 cells $\mathcal{C}_{ab}$, where $a, b \in \{0, \dots, 9\}$ index layout and process strata, respectively. One independent conditional trajectory is drawn per cell and uniformly permuted across slots. Appendix D.1 gives the discrete construction.

Process coordinate and noise replacement. Let $r = L - |\mathcal{M}|$ be the number of free residues. Let $\mathbf{X}_t^{\text{clean}} \in \mathbb{R}^{r\times 3}$ be the processed clean free-residue Cα coordinates before the translational draw. For $\mathbf{C}_r = \mathbf{I}_r - \mathbf{1}\mathbf{1}^\top/r$, define, when the denominator is nonzero,

$$\boldsymbol{v}_t = \frac{\operatorname{vec}(\mathbf{C}_r\mathbf{X}_t^{\text{clean}})}{\|\mathbf{C}_r\mathbf{X}_t^{\text{clean}}\|_F}, \qquad S = \sum_{t=\tau+1}^{t_{\text{g}}} w_t \boldsymbol{v}_t^\top \boldsymbol{\epsilon}_t, \quad w_t = (t_{\text{g}} - \tau)^{-1/2}, \tag{9}$$

where $\boldsymbol{\epsilon}_t$ is standardized translational noise. An independent unit direction handles degenerate geometry. Because $\boldsymbol{v}_t$ uses only realized history, not the cell label, target s, or future noise, $S \sim \mathcal{N}(0,1)$ under $\mathbb{P}_{\text{ref}}$ and is independent of the augmented layout.

Let Φ be the standard normal CDF. For process stratum b, draw $s = \Phi^{-1}(U)$ with $U \sim \text{Unif}(b/10, (b+1)/10)$ and independently draw $\boldsymbol{g} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. With $\boldsymbol{w} = (w_{t_{\text{g}}}, \ldots, w_{\tau+1})^\top$, set

$$\boldsymbol{\eta} = \boldsymbol{g} + \boldsymbol{w}(s - \boldsymbol{w}^\top \boldsymbol{g}), \qquad \boldsymbol{\epsilon}_t = \boldsymbol{\xi}_t + \boldsymbol{v}_t(\eta_t - \boldsymbol{v}_t^\top \boldsymbol{\xi}_t). \tag{10}$$

where $\boldsymbol{\xi}_t$ is the native standardized draw. This enforces $S = s$, retaining the native reverse-step location and scale parameters, the orthogonal innovation, and the rotational draw at each realized state, without extra denoiser evaluations. Mixing the conditional paths over $s \sim \mathcal{N}(0,1)$ recovers the matched-reference translational innovation law.

Exact paths and coverage. Under ideal arithmetic, independent random sources, and an almost surely defined sampler and evaluation:

Theorem 1 (Conditional paths and marginal preservation) *The path drawn in cell $\mathcal{C}_{ab}$ has law $\mathbb{P}_{\text{ref}}(\cdot \mid \mathcal{C}_{ab})$. After the independent uniform permutation, every slot has marginal law $\mathbb{P}_{\text{ref}}$.*

For coverage, fix before sampling a map $\kappa(O) \in \{0, 1, 2, \ldots\}$, with zero denoting failure. The same map is used for all compared schemes and assigns a category from the output O alone, independently of the other outputs in its batch. Under the same reference continuation and evaluation, compare Joint EMPS with Layout-only, Process-only, and independent and identically distributed (IID) sampling. Each single-axis scheme leaves the other axis unstratified.

Theorem 2 (Successful-category coverage) *For the counts in equation 1 under this fixed map,*

$$\begin{aligned} &\mathbb{E}K_{\text{Joint}} \geq \mathbb{E}K_A \geq \mathbb{E}K_{\text{IID}}, \qquad A \in \{\text{Layout-only}, \text{Process-only}\}, \\ &\mathbb{E}S_{+,\text{Joint}} = \mathbb{E}S_{+,\text{Layout-only}} = \mathbb{E}S_{+,\text{Process-only}} = \mathbb{E}S_{+,\text{IID}}. \end{aligned} \tag{11}$$

These are matched-reference, pre-reweighting expectations over newly generated batches. Gains are strict when the added axis distinguishes a still-missable successful category, with no general ordering between the two single-axis schemes. Appendices E.2 and E give proofs, equality cases, and subset extensions. The theorem uses a fixed outputwise category map; Section 4.5 separately evaluates MotifBench's pool-dependent clusters and Score.

3.7 The full S^3-Scaffold analysis

The matched comparison in Theorem 2 precedes allocation. We now condition on S^3-Scaffold's parent pool $\mathcal{H}$, scores $\boldsymbol{a}$, and offspring counts $\boldsymbol{n}$. Let q_{ic} be the probability that a fresh continuation from $H_\tau^{(i)}$ reaches successful category c, so $q_{i,+} = \sum_{c>0} q_{ic}$. Independent fresh continuations and evaluations give

$$\mathbb{E}[S_+ \mid \mathcal{H}, \boldsymbol{a}, \boldsymbol{n}] = \sum_i n_i q_{i,+}, \tag{12}$$

$$\mathbb{E}[K_{\text{full}} \mid \mathcal{H}, \boldsymbol{a}, \boldsymbol{n}] = \sum_{c>0} \Big[1 - \prod_i (1 - q_{ic})^{n_i}\Big]. \tag{13}$$

Guidance changes parent states and continuation probabilities, rollout changes allocation, and EMPS changes the parent pool's joint law. Relative to one continuation per parent, allocation can omit unique-category parents while extra copies can recover reachable categories. Appendix F separates these effects and derives finite-batch lower bounds on expected coverage under anchor-retention and successful-continuation conditions. Under common conditions, these bounds satisfy Joint $\geq$ Layout-only $\geq$ IID. It also relates fixed categories to set-dependent Foldseek clustering, while Appendix B.5.2 connects local guidance interventions to terminal utility.

Table 1: Comparison with selected public MotifBench submissions, ordered by Score. Shading marks S^3-Scaffold and its direct base.

Method	Score ↑	Solved /30 ↑	Successful clusters ↑
RFdiffusion3 (Butcher et al., 2025)	43.52	27.3	330.7
Protpardelle-1c (cc91-large) (Lu et al., 2025)	33.81	26	186
Genie3 (Lin et al., 2026)	31.14	23.3	298.7
S^3-Scaffold (ours)	30.27	22	225
La-Proteina (Geffner et al., 2026)	29.75	24.3	149.3
ODesign-rigid (Zhang et al., 2025)	29.39	27	161
Protpardelle-1c (Lu et al., 2025)	27.83	25	142
Proteina (Geffner et al., 2025)	25.46	17	308
RFdiffusion (base) (Watson et al., 2023)	21.95	17	205
Genie2 (Lin et al., 2024)	20.83	15	234
FrameFlow (Yim et al., 2024)	18.71	15.0	180.7

Public baseline values follow the MotifBench leaderboard and linked evaluation archives. RFdiffusion3, Genie3, La-Proteina, and FrameFlow report means over three independent runs; the other public baselines report one submitted pool.

Table 2: Component ablations with fixed RFdiffusion weights. S^3-Scaffold w/o EMPS combines gated gradient guidance and rollout reweighting. Bold marks the best value in each column.

Configuration	Score ↑	Solved /30 ↑	S_+ /3,000 ↑	Successful clusters ↑
RFdiffusion	21.95	17	481	205
Gated gradient guidance	23.24	18	506	216
Rollout reweighting	26.30	18	626	197
EMPS	23.50	17	481	212
S^3-Scaffold w/o EMPS	28.83	**22**	**684**	214
S^3-Scaffold (ours)	**30.27**	**22**	678	**225**

The RFdiffusion row retains the public Score; all other Scores are exact subset expectations for the corresponding saved pools. Each configuration contains 3,000 evaluated designs.

4 Experiments

4.1 Experimental setup

We evaluate frozen RFdiffusion on all 30 MotifBench tasks, generating 100 backbones per task and testing each with eight fresh ProteinMPNN sequences and ESMFold refolds under Section 3.1's success criterion (Dauparas et al., 2022; Lin et al., 2023; Zheng et al., 2025). MotifBench assigns taskwise Foldseek clusters in each full pool, inherits these batch-dependent labels for subset scoring, and calls a task solved when its pool contains a success. We use 50 denoising steps, with $t_{\mathrm{g}} = 25$, $t_{\mathrm{gate}} = 16$, $\tau = 10$, and EMPS at $t = 25, \ldots, 11$. Appendix A.1 lists other settings. We report Score, solved tasks, successful designs S_+, and successful clusters. The official Score samples 50 of 100 designs per task after full-pool clustering. For each in-house pool, we compute the expectation of the same statistic exactly, while retaining public baselines' reported scores. All configurations match final-candidate and endpoint-evaluation budgets (Appendix A.6). The resulting scores are exact conditional expectations for each generated pool (Appendix A.5).

4.2 Overall performance

Without updating RFdiffusion's pretrained weights, S^3-Scaffold raises Score from 21.95 to 30.27, an 8.32-point (37.9%) gain; solved tasks increase from 17 to 22, and successful clusters from 205 to 225 (Table 1). S^3-Scaffold exceeds La-Proteina and ODesign-rigid in Score and successful clusters, although both methods solve more tasks. Genie3, Protpardelle-1c (cc91-large), and RFdiffusion3 remain ahead in Score.

4.3 Component ablations

In the evaluated pools, gated gradient guidance improves yield and coverage, whereas rollout reweighting primarily improves yield. Together, gated gradient guidance and rollout reweighting produce 684 successful designs and solve 22 tasks, including three solved by neither component alone, supporting their complementary effects on difficult tasks. At the same observed $S_+ = 481$, EMPS increases RFdiffusion's successful clusters from 205 to 212. After guidance and rollout, it raises Score and clusters despite reducing S_+ from 684 to 678 (Table 2).

4.4 Mechanism evidence for successful design

Geometric control. All 481 baseline successes have generated-motif NCACO RMSD below 1 Å; none of the 1,101 designs at or above this threshold succeeds (Figure 2(a)). This strong association in the evaluated pool motivates direct geometric control. Restoring the geometric derivatives preserves RFdiffusion's forward values. Across 1,446 translation and rotation coordinates, the restored gradients agree closely with finite differences, with maximum angular error $0.625°$ and gradient norm ratios 0.99859–1.00737 (Appendix B.6). At fixed incoming states from earlier ungated trajectories, the resulting update reduces raw clean motif NCAC RMSD in 31 of 32 positive-budget probes (Figure 2(b)). These diagnostics validate gradient fidelity and frequent local improvement, while the ablations quantify endpoint effects.

Gating. In a paired five-task pilot, ungated guidance increases motif-recovery passes from 9 to 15 but decreases scaffold-consistency passes from 66 to 53, showing that motif correction can disrupt downstream compatibility. Across separate full-benchmark pools, gated variants improve Score, S_+, and successful clusters both with and without rollout, although ungated guidance alone solves one additional task. Together with the paired trade-off, these comparisons support error-dependent rather than uniformly applied guidance (Appendix G.2).

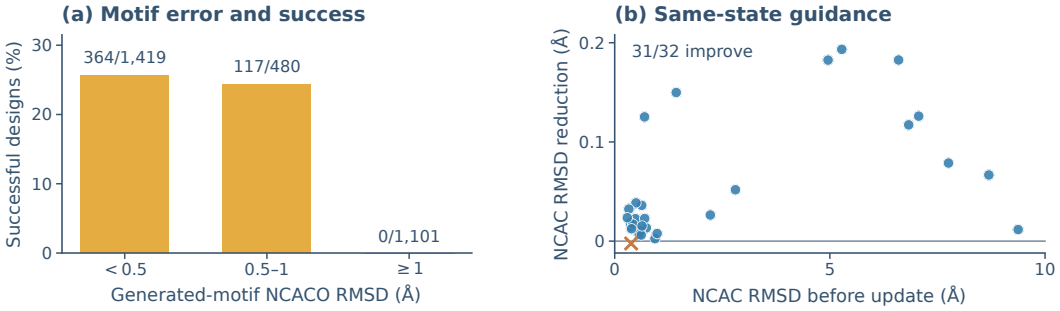

Figure 2: Geometric-control diagnostics: (a) joint success by generated-motif NCACO RMSD across 3,000 baseline designs; (b) same-state raw-clean motif NCAC RMSD reduction at 32 probes from eight ungated trajectories, with the cross marking an increase. Protocols: Appendix G.5.

Downstream compatibility. In an earlier ungated-guidance pool, 52 of 534 designs passing generated-geometry, same-sequence scaffold-consistency, and motif-retention checks still fail original-target recovery (Appendix C.1). This motivates scoring scaffold consistency and refolded-motif errors jointly. In the evaluated pools, the rollout configurations contain 626 successful designs on RFdiffusion and 684 with gated gradient guidance, compared with 481 and 506 for the respective configurations without rollout (Table 2; Appendix C.5).

4.5 Successful-structure coverage with EMPS

We measure empirical coverage using MotifBench's successful clusters, defined separately within each full 100-design task pool. The subset expectations condition on these saved pools and their cluster assignments.

On RFdiffusion alone, EMPS reaches a Score of 23.50 without additional model evaluations. The two saved pools contain the same observed number of successful designs, $S_+ = 481$, while EMPS increases full-pool successful clusters from 205 to 212. For uniform 50-design subsets from each task pool, expected total cluster coverage rises from 116.54 to 123.44, an absolute gain of 6.90 clusters (5.9%; Figure 3(a)).

Relative to S^3-Scaffold w/o EMPS, S^3-Scaffold raises expected 50-design coverage from 138.54 to 147.64 and full-pool successful clusters from 214 to 225, while increasing Score from 28.83 to 30.27 and retaining 22 solved tasks. Thus successful structures occupy more clusters even as S_+ changes from 684 to 678. Appendix G.5 relates the subset readout to observed cluster occupancies, and Appendix G.4 illustrates the joint use of the two stratification axes on one selected task.

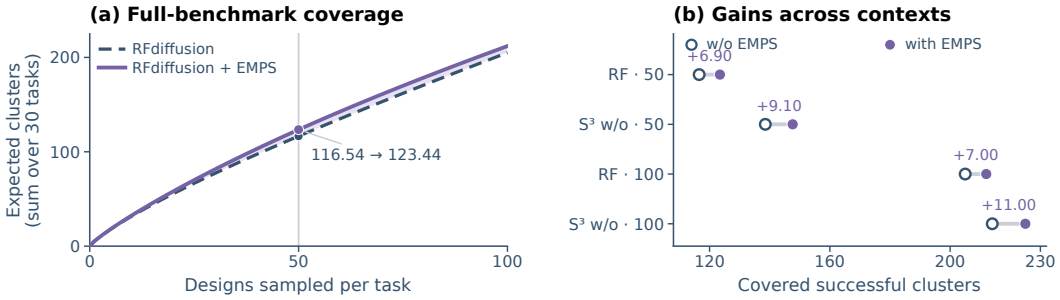

Figure 3: Empirical successful-cluster coverage in saved pools: (a) exact expected coverage for uniform subsets of RFdiffusion and RFdiffusion+EMPS pools across 30 tasks; (b) expected 50-design subset coverage and full-pool clusters before and after adding EMPS to RFdiffusion (RF) or S^3-Scaffold w/o EMPS. Each pool contains 100 designs, with Foldseek labels assigned from its full successful set and held fixed during subsampling. Panel (a) varies subset size within each fixed pool. Calculation: Appendix G.5.

5 Discussion

Successful-design yield and successful-structure coverage are related but non-interchangeable objectives. Gated gradient guidance and rollout reweighting improve yield at different stages: the former corrects motif geometry during denoising, while the latter uses sequence-design and refolding feedback to favor compatible continuations. EMPS coordinates finite-batch exploration; its coverage gains at the same observed success count and after composition show that yield alone does not ensure structural breadth.

These roles align with theory: controlled guidance decreases a local geometric objective, rollout benefits allocation under positive score–success covariance, and EMPS improves expected fixed-category coverage while preserving matched-reference marginals. Together, they provide a staged inference-time strategy for generation-time geometry, downstream compatibility, and batch diversity.

Our 30-task evaluation uses the ProteinMPNN–ESMFold pipeline; repeated generations, alternative predictors, and experimental assays can assess robustness and biological utility.

6 Conclusion

We introduced S^3-Scaffold, a training-free RFdiffusion sampling framework addressing motif geometry, downstream sequence–structure compatibility, and finite-batch exploration. Across all 30 MotifBench tasks, it raises Score, solved tasks, and successful clusters from $21.95/17/205$ to $30.27/22/225$, showing that inference-time control can improve both successful-design yield and structural coverage without retraining the generator.

AI USE STATEMENT

We used the generative AI functionality of TRAE CLI to draft and edit wording for grammar, clarity, and readability. TRAE CLI was not used to generate synthetic data or experimental results; develop theoretical models or conceptual frameworks; formulate mathematical claims; provide critical ingredients for, or assist in writing, mathematical proofs; propose or refine research hypotheses; design or provide feedback on research methodology or experiments; implement the proposed methods; clean or reformat datasets; conduct qualitative or thematic data analysis; or interpret experimental results. All scientific ideas, methods, mathematical arguments, experimental designs, data, results, and conclusions were developed and verified by the authors. The authors reviewed and revised all AI-assisted text against the underlying derivations, code, experimental records, and primary references, and take responsibility for the final content of this work, including any text produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Section 4.1 specifies the evaluation protocol and principal sampling settings. Appendix A provides S^3-Scaffold's complete algorithm, implementation settings, software versions, random-stream roles and recording protocol, benchmark-scoring procedure, and computational budgets. Appendices B, C, D, E, and F state the assumptions and provide the derivations and proofs supporting S^3-Scaffold. Appendix G reports the complete task-level results and coverage analyses, and Appendix G.5 documents the diagnostic and plotting procedures.

REFERENCES

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

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 843–852, 2023. URL https://arxiv.org/abs/2302.07121.

Jasper Butcher, Rohith Krishna, Raktim Mitra, Rafael I. Brent, Yanjing Li, Nathaniel Corley, Paul T. Kim, Jonathan Funk, Simon Mathis, Saman Salike, Aiko Muraishi, Helen Eisenach, Tuscan Rock Thompson, Jie Chen, Yuliya Politanska, Enisha Sehgal, Brian Coventry, Odin Zhang, Bo Qiang, Kieran Didi, Max Kazman, Frank DiMaio, and David Baker. De novo design of all-atom biomolecular interactions with RFdiffusion3. *bioRxiv*, 2025. doi: 10.1101/2025.09.18.676967. URL https://doi.org/10.1101/2025.09.18.676967.

Longxing Cao, Inna Goreshnik, Brian Coventry, James Brett Case, Lauren Miller, Lisa Kozodoy, Rita E. Chen, Lauren Carter, Alexandra C. Walls, Young-Jun Park, Eva-Maria Strauch, Lance Stewart, Michael S. Diamond, David Veesler, and David Baker. De novo design of picomolar SARS-CoV-2 miniprotein inhibitors. *Science*, 370(6515):426–431, 2020. doi: 10.1126/science.abd9909. URL https://doi.org/10.1126/science.abd9909.

Jacob K. Christopher, Austin Seamann, Jingyi Cui, Sagar Khare, and Ferdinando Fioretto. Constrained diffusion for protein design with hard structural constraints. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=kkvqVRu2Zy.

Gabriele Corso, Yilun Xu, Valentin De Bortoli, Regina Barzilay, and Tommi S. Jaakkola. Particle guidance: Non-I.I.D. diverse sampling with diffusion models. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=KqbCvIFBY7.

Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J. Ragotte, Lukas F. Milles, Basile I. M. Wicky, Alexis Courbet, Rob J. de Haas, Neville Bethel, Philip J. Y. Leung, Timothy F.

Huddy, Sam Pellock, Doug Tischer, Frederick Chan, Brian Koepnick, Hahn Nguyen, Alex Kang, Banumathi Sankaran, Asim K. Bera, Neil P. King, and David Baker. Robust deep learning–based protein sequence design using ProteinMPNN. *Science*, 378(6615):49–56, 2022. doi: 10.1126/science.add2187. URL https://doi.org/10.1126/science.add2187.

Pierre Del Moral. *Feynman–Kac Formulae: Genealogical and Interacting Particle Systems with Applications*. Probability and Its Applications. Springer, 2004. doi: 10.1007/978-1-4684-9393-1. URL https://doi.org/10.1007/978-1-4684-9393-1.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, volume 34, pp. 8780–8794, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/49ad23d1ec9fa4bd8d77d02681df5cfa-Abstract.html.

Kieran Didi, Francisco Vargas, Simon V. Mathis, Vincent Dutordoir, Emile Mathieu, Urszula J. Komorowska, and Pietro Liò. A framework for conditional diffusion modelling with applications in motif scaffolding for protein design. In *NeurIPS 2023 Workshop on New Frontiers of AI for Drug Discovery and Development*, 2023. URL https://arxiv.org/abs/2312.09236.

Tomas Geffner, Kieran Didi, Zuobai Zhang, Danny Reidenbach, Zhonglin Cao, Jason Yim, Mario Geiger, Christian Dallago, Emine Kucukbenli, Arash Vahdat, and Karsten Kreis. Proteina: Scaling flow-based protein structure generative models. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=TVQLu34bdw. Oral.

Tomas Geffner, Kieran Didi, Zhonglin Cao, Danny Reidenbach, Zuobai Zhang, Christian Dallago, Emine Kucukbenli, Karsten Kreis, and Arash Vahdat. La-Proteina: Atomistic protein generation via partially latent flow matching. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=RDerF20JYT.

Paul Glasserman, Philip Heidelberger, and Perwez Shahabuddin. Asymptotically optimal importance sampling and stratification for pricing path-dependent options. *Mathematical Finance*, 9 (2):117–152, 1999. doi: 10.1111/1467-9965.00065. URL https://doi.org/10.1111/1467-9965.00065.

Erik Hartman, Jonas Wallin, Johan Malmström, and Jimmy Olsson. Controllable protein design with particle-based Feynman–Kac steering. *arXiv preprint arXiv:2511.09216*, 2025. URL https://arxiv.org/abs/2511.09216.

Jonathan Ho, Tim Salimans, Alexey A. Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 8633–8646, 2022. URL https://arxiv.org/abs/2204.03458.

Michael Jendrusch and Jan O. Korbel. Efficient protein structure generation with sparse denoising models. *Nature Machine Intelligence*, 7(9):1429–1445, 2025. doi: 10.1038/s42256-025-01100-z. URL https://doi.org/10.1038/s42256-025-01100-z.

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=vi3DjUhFVm. Spotlight.

Anna Lauko, Samuel J. Pellock, Kiera H. Sumida, Ivan Anishchenko, David Juergens, Woody Ahern, Jihun Jeung, Alexander F. Shida, Andrew Hunt, Indrek Kalvet, Christoffer Norn, Ian R. Humphreys, Cooper Jamieson, Rohith Krishna, Yakov Kipnis, Alex Kang, Evans Brackenbrough, Asim K. Bera, Banumathi Sankaran, K. N. Houk, and David Baker. Computational design of serine hydrolases. *Science*, 388(6744):eadu2454, 2025. doi: 10.1126/science.adu2454. URL https://doi.org/10.1126/science.adu2454.

Yeqing Lin, Minji Lee, Zhao Zhang, and Mohammed AlQuraishi. Out of many, one: Designing and scaffolding proteins at the scale of the structural universe with Genie 2. *arXiv preprint arXiv:2405.15489*, 2024. URL https://arxiv.org/abs/2405.15489.

Yeqing Lin, Minji Lee, Aakarsh Vermani, Ellena Jiang, Sebastiaan De Cooman, Matej Špeťko, and Mohammed AlQuraishi. Fast and ultra-capable protein design: Advancing the frontier through atomistic SE(3)-equivariance with Genie 3. *bioRxiv*, 2026. doi: 10.64898/2026.05.01.722168. URL https://doi.org/10.64898/2026.05.01.722168.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023. doi: 10.1126/science.ade2574. URL https://doi.org/10.1126/science.ade2574.

Tianyu Lu, Richard Shuai, Petr Kouba, Zhaoyang Li, Yilin Chen, Akio Shirali, Jinho Kim, and Po-Ssu Huang. Conditional protein structure generation with Protpardelle-1c. *bioRxiv*, 2025. doi: 10.1101/2025.08.18.670959. URL https://doi.org/10.1101/2025.08.18.670959.

Michael D. McKay, Richard J. Beckman, and William J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979. doi: 10.1080/00401706.1979.10489755. URL https://doi.org/10.1080/00401706.1979.10489755.

Mashrur M. Morshed and Vishnu Boddeti. DiverseFlow: Sample-efficient diverse mode coverage in flows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23303–23312, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Morshed_DiverseFlow_Sample-Efficient_Diverse_Mode_Coverage_in_Flows_CVPR_2025_paper.html.

Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=9_gsMA8MRKQ.

Brian L. Trippe, Jason Yim, Doug Tischer, David Baker, Tamara Broderick, Regina Barzilay, and Tommi S. Jaakkola. Diffusion probabilistic modeling of protein backbones in 3D for the motif-scaffolding problem. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2206.04119.

Gal Vinograd, Idan Achituve, and Ethan Fetaya. Diverse sampling in diffusion models with marginal preserving particle guidance. *arXiv preprint arXiv:2605.06553*, 2026. URL https://arxiv.org/abs/2605.06553.

Chuanzhen Wang, Meade Cleti, and Pete Jano. ProHiFlo: Hierarchical flow matching with functional guidance for de novo protein generation. *arXiv preprint arXiv:2606.11243*, 2026. URL https://arxiv.org/abs/2606.11243.

Jue Wang, Sidney Lisanza, David Juergens, Doug Tischer, Joseph L. Watson, Karla M. Castro, Robert Ragotte, Amijai Saragovi, Lukas F. Milles, Minkyung Baek, Ivan Anishchenko, Wei Yang, Derrick R. Hicks, Marc Expòsit, Thomas Schlichthaerle, Jung-Ho Chun, Justas Dauparas, Nathaniel Bennett, Basile I. M. Wicky, Andrew Muenks, Frank DiMaio, Bruno Correia, Sergey Ovchinnikov, and David Baker. Scaffolding protein functional sites using deep learning. *Science*, 377(6604):387–394, 2022. doi: 10.1126/science.abn2100. URL https://doi.org/10.1126/science.abn2100.

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

Luhuan Wu, Brian L. Trippe, Christian A. Naesseth, David M. Blei, and John P. Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. In *Advances in*

Neural Information Processing Systems, volume 36, pp. 31372–31403, 2023. URL https://arxiv.org/abs/2306.17775.

Junde Xu, Yuansheng Huang, Zijun Gao, Lihang Liu, Xiaoming Fang, Yu Kang, Jiezhong Qiu, and Pheng Ann Heng. Exploring the alignment of generation and understanding in protein structure modeling. *arXiv preprint arXiv:2607.13503*, 2026. URL https://arxiv.org/abs/2607.13503.

Yilun Xu, Mingyang Deng, Xiang Cheng, Yonglong Tian, Ziming Liu, and Tommi S. Jaakkola. Restart sampling for improving generative processes. In *Advances in Neural Information Processing Systems*, volume 36, pp. 76806–76838, 2023. URL https://openreview.net/forum?id=wFuemocyHZ.

Jason Yim, Andrew Campbell, Emile Mathieu, Andrew Y. K. Foong, Michael Gastegger, José Jiménez-Luna, Sarah Lewis, Victor Garcia Satorras, Bastiaan S. Veeling, Frank Noé, Regina Barzilay, and Tommi S. Jaakkola. Improved motif-scaffolding with SE(3) flow matching. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=fa1ne8xDGn.

James Matthew Young and O. Deniz Akyildiz. On diffusion posterior sampling via sequential Monte Carlo for zero-shot scaffolding of protein motifs. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=KXRYY7iwqh.

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

Zhuoqi Zheng, Bo Zhang, Kieran Didi, Kevin K. Yang, Jason Yim, Joseph L. Watson, Hai-Feng Chen, and Brian L. Trippe. MotifBench: A standardized protein design benchmark for motif-scaffolding problems. *arXiv preprint arXiv:2502.12479*, 2025. URL https://arxiv.org/abs/2502.12479.

Zhuoqi Zheng, Bo Zhang, Bozitao Zhong, Jinyu Yu, Kexin Liu, Zhengxin Li, Junjie Zhu, Ting Wei, and Hai-Feng Chen. Scaffold-Lab: Critical evaluation and ranking of protein backbone generation methods in a unified framework. *PLOS Computational Biology*, 22(5):e1014290, 2026. doi: 10.1371/journal.pcbi.1014290. URL https://doi.org/10.1371/journal.pcbi.1014290.

Shaowen Zhu, Siddhant Gulati, Yuxuan Liu, Siddhi Kotnis, Qing Sun, and Yang Shen. Multimodal diffusion for joint design of protein sequence and structure. *Protein Science*, 34(12):e70340, 2025. doi: 10.1002/pro.70340. URL https://doi.org/10.1002/pro.70340.

Appendix guide. Appendix A specifies the end-to-end algorithm, execution semantics, fixed settings, evaluation protocol, random streams, and computational budgets. Appendices B, C, and D give the geometric derivatives, rollout allocation, and EMPS construction with their mechanism proofs and implementation diagnostics. Appendix E analyzes finite-budget successful-category coverage under matched pre-allocation sampling. Appendix F connects the components through conditional endpoint identities, an exact allocation decomposition, and coverage lower bounds. Appendix G reports the complete experimental evidence, plotting protocols, and public-baseline provenance.

A Complete algorithm and reproducibility

A.1 End-to-end sampling algorithm

Algorithm 1 specifies S^3-Scaffold for one motif-scaffolding task. The models remain frozen throughout sampling and evaluation. The input includes the motif coordinates, legal contig and length constraints, and the task's fixed amino-acid positions. With $\tau = 10$, we distinguish a saved parent state $H_\tau^{(i)}$ from the clean structure $\mathbf{P}_i$ used to score it. The function Step applies the configured gated gradient guidance and then one native denoising transition. Its final argument is an optional EMPS descriptor: a descriptor activates the EMPS innovation only in its prescribed window, whereas NONE leaves the native translational innovation unchanged. Its ordering is specified below.

Algorithm 1 S^3-Scaffold: gated gradient guidance, rollout reweighting, and EMPS.

Require: Motif and scaffold constraints; frozen RFdiffusion, ProteinMPNN, and ESMFold; separate random streams for each sampling role.

Ensure: $M = 100$ final backbones with eight fresh sequence-design and refolding evaluations each.

Construct the 100 EMPS descriptors with Algorithm 3.
for parent slot $i = 1, \ldots, M$ **do**
  Initialize H using its assigned layout and native initialization stream.
  for $t = 50, 49, \ldots, 1$ **do**
    if $t = 10$ **then**
      $H_\tau^{(i)} \leftarrow \mathrm{Snapshot}(H)$ ▷ before guidance and denoising
    end if
    $(H, \mathbf{P}) \leftarrow \mathrm{Step}(H, t, \text{EMPS descriptor}_i)$.
    if $t = 10$ **then**
      $\mathbf{P}_i \leftarrow \mathbf{P}$ ▷ the production clean prediction at this step
    end if
  end for
  Sample four sequences for $\mathbf{P}_i$ and refold each with ESMFold.
  Compute a_i from the paired errors r_{ij}, using the stable energy $E_i = -\log a_i$.
end for
Compute π_i using equation 21; set $C_i = \sum_{r=1}^{i} \pi_r$.
Draw $u \sim \mathrm{Unif}[0, 1/M)$ independently.
for offspring slot $k = 1, \ldots, M$ **do**
  $i \leftarrow \min\{r : C_r > u + (k-1)/M\}$.
  Reconstruct the parent's layout and motif mapping; restore $H_\tau^{(i)}$ into a separate state H.
  Install the fresh continuation stream assigned to offspring slot k.
  for $t = 10, 9, \ldots, 1$ **do**
    $(H, \mathbf{P}) \leftarrow \mathrm{Step}(H, t, \textsc{None})$.
  end for
  Set final backbone $\mathbf{X}_k \leftarrow \mathbf{P}$.
  Sample eight new sequences using slot k's endpoint stream; refold and evaluate them.
end for
return the 100 evaluated final designs.

A.2 EXECUTION SEMANTICS

The allocation boundary. The snapshot is taken immediately before the $t = 10$ step. The subsequent step produces $\mathbf{P}_i$, including any enabled guidance and native clean-structure processing. The saved snapshot retains the incoming coordinates, self-conditioning history, and guidance records throughout this step. Each selected offspring restores that snapshot and executes $t = 10$ again with fresh continuation randomness. The scoring structure is the returned all-atom clean prediction, including its predicted motif coordinates.

The frozen capture implementation also completes the parent trajectory through $t = 1$ and records that endpoint for diagnostics. Algorithm 1 includes those steps to match the executed computation. Parent endpoints serve as diagnostic structures outside the eight-sequence endpoint evaluation and carry no endpoint-success labels. Allocation uses the intermediate scoring structures, and the final batch contains the 100 offspring endpoints.

Ordering within one denoising step. First compute and cache the effective guidance budget from the time schedule and the latest finite recorded motif loss. When this budget is positive, evaluate the translation and rotation derivatives separately at the same input coordinates, sequence, and self-conditioning history. Mask motif controls, normalize the two directions, and choose the common feasible scale with Algorithm 2. The search accepts scales according to coordinate displacement within the rotation-limited interval. Derivative calls preserve the incoming self-conditioning and random state. The production step then uses the updated coordinates with that preserved history and native random state. A refused derivative geometry check leaves the coordinates unchanged for the native step.

The native forward pass first produces the raw clean prediction and updates self-conditioning. It then reconstructs all atoms. For $t > 1$, native motif alignment precedes the translational and rotational draws. EMPS acts at the translational draw through Algorithm 4; its direction uses this production clean prediction after the geometric update. Rotational sampling and the remaining native state updates follow. At $t = 1$, the native sampler directly returns the clean structure without posterior noise.

Saved state and fresh continuation. A snapshot contains coordinates, sequence, the previous prediction, guidance ledger and head state, and bindings for the layout, motif mask, and residue mapping. It also records the Python, NumPy, Torch CPU, and CUDA random states. The chosen RFdiffusion configuration has no additional active recurrent MSA, pair, or state tensors. Reconstructing a continuation performs the native initialization and warmup before restoring the saved state. Only after restoration does the implementation reseed the continuation generators. Thus two offspring of one parent share its sampling state and conditioning, while using different future noise streams. Future schedule values are computed from the restored ledger and cached on first use at each step.

The paired score measurements and stable weight calculation are specified in Appendix C.2.

A.3 FIXED SETTINGS AND COMPONENT SWITCHES

Table 3 collects the settings used by S^3-Scaffold. Task-specific motif mappings and legal scaffold lengths are supplied by the benchmark tasks. Sequence design fixes the amino-acid positions declared by each task, mapped into its sampled layout. ProteinMPNN uses its full-backbone model, and all four scoring or eight endpoint sequences are evaluated directly in their sampling order. ESMFold uses its native inference settings, with no recycle or chunk-size override.

The components define separate sampling operations. Omitting rollout reweighting removes the four-sequence scoring stage and assigns one final output to each trajectory. The RFdiffusion-with-EMPS and layout/process ablation runs execute all 50 native denoising steps without interruption or a capture–resume boundary. The one-fresh-continuation reference used in Appendix E.1 has the same conditional tail law under its independent-random-tape assumptions. Omitting gated gradient guidance leaves the state unperturbed before each native denoising step. Omitting EMPS uses the native layout prior and native translational innovations. The remaining operations and final evalua-

Table 3: Fixed settings for S^3-Scaffold. All neural-network parameters remain frozen.

Setting	Value
Base generator and evaluation	
RFdiffusion	Base checkpoint; nonsymmetric SelfConditioning; evaluation mode; float32 production
Denoising schedule	$T = 50$, terminal step $t = 1$; no partial diffusion or native guiding potentials
Candidate budget	100 parents and 100 final offspring per task
Sequence-design budget	Four sequences per scoring structure; eight fresh sequences per final backbone
Sequence and folding models	ProteinMPNN `v_48_020`, sampling temperature 0.1, backbone noise 0; ESMFold v1
Gated gradient guidance	
Guidance schedule	Zero budget at $t = 25$; ramp to 0.3 Å by $t = 10$, constant base budget thereafter
Guidance gate	Active at $t \leq 16$; midpoint 0.75 Å, softness 0.15 Å
Control search	Maximum residue rotation 0.5 radians; 20 displacement-budget bisections
Rollout reweighting	
Boundary and mixture weight	Snapshot before $t = 10$; $\lambda = 0.25$
Score normalization	2 Å scaffold error and 1 Å refolded-motif RMSD; exponential temperature 1
EMPS	
EMPS grid and window	10×10; $t = 25, \ldots, 11$; $w_t = 1/\sqrt{15}$
EMPS arithmetic	Float64 direction and projection; replacement cast to production dtype

tion stay as specified for each comparison. These definitions make the individual components and their combinations explicit; empirical comparisons use their declared configurations.

Status of the settings. The reported full-benchmark pools use the same values in Table 3 for every task. The RFdiffusion schedule and downstream evaluation follow the base generator and MotifBench protocol. The 10×10 EMPS grid matches the 100-candidate batch, with one trajectory assigned to each joint cell. Its controlled window starts at the shared schedule boundary $t = 25$ and ends immediately before the rollout boundary; the guidance budget is zero at $t = 25$, so the first positive-budget guidance update is at $t = 24$. The rollout score reuses the endpoint thresholds only to normalize its two errors to comparable scales. The allocation identity in equation 8 applies to any finite positive parent scores. The remaining guidance and allocation constants are fixed implementation hyperparameters that define the evaluated procedure.

Development-stage setting evidence. Before introducing the error gate and EMPS, we compared rollout boundaries $\tau \in \{5, 8, 10\}$ under the same frozen ungated-guidance profile, rollout implementation, sequence budgets, and endpoint criterion. Table 4 reports the complete 30-task records. The pools use independent random streams, and the number of child denoising steps changes with τ; this development comparison therefore covers separate pools with different computational budgets. All 30 benchmark tasks participated in development; no held-out tuning split was used.

Table 4: Development-stage rollout-boundary comparison before gated guidance and EMPS. Each row contains 30 tasks and 3,000 final designs. Score is the archived official 1,000-resample estimate; S_+ is the number of successful designs, and clusters are summed within task. Bold marks the largest value in each column.

Boundary τ	Score	Solved	S_+	Clusters
5	25.6845	**20**	590	194
8	24.8338	18	**604**	198
10	**26.4548**	**20**	597	**202**

The $\tau = 10$ pool has the highest Score and cluster count, ties the largest solved-task count, and is second in successful-design count. We retained $\tau = 10$ for the final reported full-benchmark rollout configurations. The reported full-benchmark EMPS configurations likewise end their process window at $t = 11$, immediately before this boundary. The archived execution order was $\tau = 5$, $\tau = 10$, adoption of $\tau = 10$, and then the independent $\tau = 8$ run; the last comparison supplies retrospective support for the earlier selection. The available boundary sweep covers this ungated development configuration. The $\tau = 5$ and $\tau = 10$ rows aggregate completed task cohorts, whereas the $\tau = 8$ row was produced in one 30-task campaign.

The maximum guidance budget and ramp were set in a separate preregistered development screen of 176 trajectories on two tasks. That screen crossed the derivative backend, budgets of 0.1 and 0.3 Å, and schedule endpoints $(40, 10)$, $(25, 10)$, and $(15, 5)$, then used independent seeds for confirmation. The accepted 0.3 Å, $(25, 10)$ setting had confirmation refusal rates of 1.04% and 0.52% on the two tasks, below the prespecified 10% stability threshold. The $(40, 10)$ schedule produced a larger median geometry gain but refusal rates of 14.74% and 13.46%, which excluded it from confirmation. This screen assessed geometric response and numerical stability. The gate constants were subsequently selected using offline first-order profiles. Together, these records document how the fixed settings arose during development.

A.4 Benchmark configuration, software, and random streams

We retain the official motif fragments, allowed gap lengths, scaffold lengths, and fixed amino-acid positions for all 30 tasks. Input motif residues are mapped into each sampled contig before generation and sequence design. We retain these constraints, the residue mappings, and the input and target coordinate-file hashes. This mapping is also used when computing the refolded-motif error.

The RFdiffusion source is based on commit `86507b6538f51fce57b5a72477165f03999ed7ae`, with the geometric derivative restoration described in Appendix B.2. The Base checkpoint remains unchanged. ProteinMPNN uses `v_48_020`, and ESMFold uses its v1 model; their inference settings are listed in Table 3. The saved clustering configuration uses Foldseek `8.ef4e960`, alignment type 1, alignment mode 2, and a TM-score threshold of 0.6. We use each reported pool's saved cluster assignment when calculating its score and coverage.

All reported in-house full-benchmark pools use Python 3.9.16, NumPy 1.22.4, CUDA 12.1, cuDNN 8.9.2, and PyTorch `2.1.0+cu121` on NVIDIA H100 80GB GPUs. Matrix-multiplication TF32 is disabled; cuDNN TF32 is enabled. Production coordinates use float32, while the geometric alignment and EMPS projection calculations use the precision specified in their respective algorithms. We retain the checkpoint hash, gradient-source identities, recorded environment, and representative model-call counts.

We record case, slot, and role-specific seeds for every in-house run, distinguishing parent generation, internal sequence scoring, offspring continuation, endpoint sequence design, and the systematic allocation offset. Each task–configuration EMPS plan seed is recorded together with its plan hash. Within EMPS, separate streams generate the refined layout draws, process-coordinate targets, scalar Gaussian vectors, slot permutation, and fallback directions. Continuation and endpoint streams are indexed by offspring slot, including when multiple slots select the same parent. The single-axis runs retain the declared generation and endpoint seeds when disabling one stratification axis. The public RFdiffusion comparison uses its released design pool.

Restarting an interrupted computation retains its task, slot, and sampling plan, with retries assigned to the same design in the scientific pool. Saved PRNG checkpoints are execution metadata. The probability analyses in Appendices D.3, E, and F instead represent unconsumed randomness by independent random tapes. Their conditioning states contain every sampler variable read by the continuation policy, but exclude those unused future tapes.

Retained reproducibility records. For each in-house pool, the retained records identify its task and configuration, model and checkpoint versions, task constraints and residue mapping, role-specific seeds, final designs and endpoint evaluations, and saved clustering assignment. The records supporting the figures additionally identify the source pool or earlier diagnostic, the selected states, and the calculation applied to them. These identities distinguish full-benchmark endpoint pools from the earlier guidance diagnostics and downstream failure analyses. The bundled plotting CSVs provide selected measurements and summaries for figure regeneration.

A.5 Benchmark scoring and evaluation pools

Official subset statistic. For each task, MotifBench first clusters the successful backbones in the full 100-design pool. The cluster labels remain fixed during scoring. Its official bootstrap draws 50 distinct design indices uniformly, counts the represented successful clusters F, and evaluates $\phi(F) = 110F/(F+5)$. The released routine averages 1,000 such draws and then averages the task scores over all 30 tasks. The exact calculation below integrates the same subset statistic conditional on the saved pool and clustering. It removes Monte Carlo scoring variation while leaving variation between generated pools intact.

Exact occupancy calculation. For task d, let $n_{d1}, \ldots, n_{dK_d}$ be the successful-cluster sizes and let $n_{d0} = 100 - \sum_c n_{dc}$ count failed designs. Define the polynomial

$$G_d(x,y) = (1+x)^{n_{d0}} \prod_{c=1}^{K_d} \left[1 + y\big((1+x)^{n_{dc}} - 1\big)\right]. \tag{14}$$

The coefficient $[x^m y^f] G_d$ counts subsets containing m designs and representing exactly f successful clusters. The failure factor contributes selected designs without increasing the cluster count. Each successful-cluster factor contributes either no selected member or a nonempty subset and one represented cluster. Consequently, the exact reported score for our full-benchmark pools is

$$\mathrm{Score}_{\mathrm{exact}} = \frac{1}{30} \sum_{d=1}^{30} \sum_{f=0}^{50} \frac{110f}{f+5} \frac{[x^{50} y^f] G_d(x,y)}{\binom{100}{50}}. \tag{15}$$

We compute the coefficients by integer dynamic programming, retaining powers of x only up to 50. For every task, their sum over f is checked against $\binom{100}{50}$. We apply the nonlinear score within the expectation. The expected cluster count is reported separately in Figure 3; its calculation is given in Appendix G.5. Public baseline scores in Tables 1 and 2 retain the published values.

S^3-Scaffold evaluation pool. The reported S^3-Scaffold result uses 100 designs on each of the 30 tasks, for 3,000 evaluated designs. Tables 1 and 2 use this pool for S^3-Scaffold. Each in-house configuration in Table 2 is likewise represented by one saved 100-design pool per task. Its exact Score integrates subset selection conditional on that saved pool and clustering. The public RFdiffusion comparison and the other public baselines use their released submission pools or reported multi-run summaries. Table 1 identifies the four public methods reported as three-run means.

Recomputation records. For every exact-scored in-house task–configuration pool, the retained evaluation records include the per-design success indicator, successful-cluster assignment, and resulting cluster occupancies. Together with the failed-design count, these occupancies specify the polynomial in equation 14 and therefore permit recomputation of both the exact Score and expected subset coverage. Appendix G.5 gives the corresponding coverage calculation and identifies the source records used by the diagnostic figures.

A.6 Generation and evaluation budgets

Table 5 separates trajectory work from sequence-design and refolding budgets. A no-rollout configuration generates 100 uninterrupted trajectories per task. A rollout configuration captures 100 parents and generates 100 independently continued offspring. The executed capture code completes all 50 parent steps, including the diagnostic suffix after the saved $t = 10$ state. Each offspring then

Table 5: Budgets for the implemented 50-step protocol, per task and per 100 reported designs. Sequence counts are paired sequence-design/refolding evaluations. Production steps exclude derivative forwards and initialization warmup.

Configuration	Parent steps	Child steps	Scoring seq.	Final seq.
RFdiffusion, guidance, or EMPS	5,000	0	0	800
Rollout reweighting or S^3-Scaffold	5,000	1,000	400	800

executes the ten steps from $t = 10$ through $t = 1$. Thus the listed production-step counts include the diagnostic parent suffix.

At each positive-budget guidance step, two additional RFdiffusion forwards compute the translation and rotation derivatives. The original production forward follows the geometric update. In the recorded S^3-Scaffold 11_3TQB example, the parent executes 50 production forwards, 48 derivative forwards, and two warmup forwards, totaling 100. Its offspring executes ten production forwards, 20 derivative forwards, and two warmup forwards, totaling 32. These counters cover RFdiffusion execution; the sequence-evaluation budgets account for ProteinMPNN and ESMFold. Warmup calls restore the incoming sampling state before production begins.

EMPS reuses the current clean prediction and adds no neural-network evaluations. In the four-task execution check in Appendix D.8, each uninterrupted trajectory has 50 production forwards and two restored warmup forwards, with zero additional EMPS forwards. Its extra work consists of layout bookkeeping and Gaussian projections.

The full-benchmark S^3-Scaffold run produces 3,000 offspring designs and uses 12,000 internal scoring sequences and 24,000 final evaluation sequences. It retains one complete 100-design pool per task, as specified in Appendix A.5. These budgets count the completed scientific pools; machine scheduling and execution retries are separate from the sampling protocol. All in-house configurations are compared at the same reported final-candidate count and endpoint evaluation budget. Computational work varies by component: rollout reweighting adds parent scoring and child continuations, gated gradient guidance adds derivative forwards, and EMPS adds the non-neural bookkeeping and projections described above.

Measured execution cost. Table 6 gives the recorded phase times for the completed 30-task pools. These full-benchmark measurements used NVIDIA H100 80GB GPUs under the software environment in Appendix A.4. Trajectory times use the denoising-loop timer, which starts after process construction and warmup; scoring and endpoint times use the corresponding evaluation-stage records. The table aggregates these recorded process-stage durations. It excludes queueing and incompletely recorded failed attempts, and its trajectory columns exclude imports, model construction, warmup, and validation.

The full runs used one to eight concurrent processes per GPU, so relative-cost comparisons require stratification by concurrency. Initialization is also material: for the S^3-Scaffold w/o EMPS pool, the difference between the process and denoising-loop timers sums to 41.6 hours. Both timers begin after platform queueing.

We therefore estimate relative trajectory cost only on the five common pilot tasks after stratifying by concurrency. Table 7 reports the concurrency-one stratum; the plain reference is the measured zero-dose RFdiffusion arm. These measurements cover trajectory generation through the parent endpoint.

Table 6: Recorded phase time for completed 30-task pools, in aggregate hours. A dash denotes a phase absent from that configuration. The final column sums the unrounded phase records.

Configuration	Parent	Child	Scoring	Endpoint	Sum
RFdiffusion + EMPS	25.36	–	–	16.22	41.58
S^3-Scaffold w/o EMPS	80.89	29.55	10.45	15.88	136.76
S^3-Scaffold	91.84	33.50	10.13	16.03	151.52

Table 7: Measured per-design trajectory time on the same five tasks with one process per GPU. Relative cost is normalized by the plain RFdiffusion pilot.

Configuration	Seconds/design	Relative
RFdiffusion pilot	25.04	1.00×
RFdiffusion + EMPS	26.73	1.07×
Gated gradient guidance	92.47	3.69×
S^3-Scaffold w/o EMPS, parent	90.35	3.61×
S^3-Scaffold, parent	97.93	3.91×

The independently tabulated concurrency-two stratum gives corresponding ratios of $1.08\times$, $3.63\times$, $3.64\times$, and $3.90\times$ for the four non-reference rows, respectively. Thus derivative-based guidance is the dominant parent-trajectory cost. The recorded peak device memory rises from approximately 1.3 GB for the plain trajectory to 57.3 GB when geometric derivatives are retained, which limits usable concurrency.

At full-benchmark scale, the RFdiffusion forward counters are 395,509 for S^3-Scaffold w/o EMPS and 395,518 for S^3-Scaffold, a difference of nine calls (0.002%). Together with the explicit EMPS counters in Appendix D.8, this verifies that EMPS schedules no additional model evaluation. The EMPS-associated wall-clock differences in the controlled pilot comparisons are below 10%. These observed differences include execution variability: available campaigns differ in host and concurrency, and comparable repeated runs vary by about 7%. Complete timing records are unavailable for the archived public RFdiffusion pool and the rollout-only endpoint evaluation; full-benchmark runtime totals are reported for configurations with complete phase records.

B Faithful geometric gradients and gated gradient guidance

B.1 Geometric objective and residue-wise controls

Fix the diffusion time, sequence, task conditioning, and self-conditioning history. Let $\boldsymbol{x}_{\ell a} \in \mathbb{R}^3$ be atom a of residue ℓ, and let L be the residue count. Write $\mathcal{M}$ for the motif residues and $\mathcal{B} = \{\mathrm{N}, \mathrm{C}\alpha, \mathrm{C}\}$ for the controlled atoms. For translations $\boldsymbol{u}_\ell$ and axis-angle rotations $\boldsymbol{\omega}_\ell$, the mathematical control map is

$$[\mathcal{T}_{\boldsymbol{u},\boldsymbol{\omega}}\boldsymbol{x}]_{\ell a} = \boldsymbol{x}_{\ell a} + \boldsymbol{u}_\ell + \big(\exp([\boldsymbol{\omega}_\ell]_\times) - \mathbf{I}\big)(\boldsymbol{x}_{\ell a} - \boldsymbol{x}_{\ell,\mathrm{C}\alpha}), \qquad a \in \mathcal{B}. \tag{16}$$

Here $[\boldsymbol{\omega}]_\times \boldsymbol{v} = \boldsymbol{\omega} \times \boldsymbol{v}$. The translation and axis-angle components are expressed in the global Cartesian basis; the rotation center is the current residue Cα. Other atoms remain unchanged, and $\boldsymbol{u}_\ell = \boldsymbol{\omega}_\ell = 0$ on $\mathcal{M}$. The implementation adds the Rodrigues rotation increment to the original coordinates. At zero control, this increment is exactly zero. For squared angle $z = \|\boldsymbol{\omega}_\ell\|^2 < 10^{-6}$, its coefficients use $\sin(\sqrt{z})/\sqrt{z} \approx 1 - z/6 + z^2/120$ and $(1-\cos(\sqrt{z}))/z \approx 1/2 - z/24 + z^2/720$. The remaining branch uses the trigonometric expressions. The exponential map in equation 16 is the real-valued reference for the analysis.

The derivative target is the current raw denoiser prediction, before motif alignment and reconstruction in the sampling routine. Let $\mathbf{J}_\theta$ be its Jacobian with respect to the current coordinates, including coordinate-dependent preprocessing, with coordinate arrays vectorized for the Jacobian. For the stacked controls $\boldsymbol{z} = (\boldsymbol{u}, \boldsymbol{\omega}) \in \mathbb{R}^{6L}$, let $\mathbf{J}_{\mathrm{ctrl}}$ be the Jacobian of equation 16 at zero. With the motif selection matrix $\mathcal{S}_\mathcal{M}$ of Section 3.4, the loss derivative is assembled as

$$\nabla_{\boldsymbol{z}} \mathcal{U}_H(0,0) = \mathbf{J}_{\mathrm{ctrl}}^\top \mathbf{J}_\theta^\top \mathrm{vec}\big(\mathcal{S}_\mathcal{M}^\top \nabla_{\mathbf{Z}} \ell_{\mathbf{R}_*}\big),$$

where $\mathcal{U}_H$ and $\ell_{\mathbf{R}_*}$ denote the fixed-alignment losses defined below. In coordinate form, if $\boldsymbol{b}_{\ell a}$ is the loss derivative with respect to $\boldsymbol{x}_{\ell a}$, the two control derivatives are

$$\nabla_{\boldsymbol{u}_\ell} \mathcal{U}_H = \sum_{a \in \mathcal{B}} \boldsymbol{b}_{\ell a}, \qquad \nabla_{\boldsymbol{\omega}_\ell} \mathcal{U}_H = \sum_{a \in \mathcal{B}} (\boldsymbol{x}_{\ell a} - \boldsymbol{x}_{\ell,\mathrm{C}\alpha}) \times \boldsymbol{b}_{\ell a}.$$

These expressions apply to free residues $\ell \notin \mathcal{M}$; the projected motif components are zero. Thus translation and rotation expose distinct geometric degrees of freedom at each free residue.

B.2 Faithful restoration of RFdiffusion geometric derivatives

Torsion features. The original preprocessing replaces undefined torsion components with constants. We record which components are finite before this replacement. A second evaluation uses finite placeholder coordinates to compute derivatives. Only the originally valid components contribute to the derivative; the original feature values and masks supply the forward pass. Writing $\mathbf{V}$ for the component-validity mask, this takes the form

$$a_{\mathrm{tor}} = \mathrm{sg}(a_{\mathrm{tor,ref}}) + \mathbf{V} \odot \big(a_{\mathrm{tor,geo}} - \mathrm{sg}(a_{\mathrm{tor,geo}})\big).$$

Missing-atom components therefore retain their original constants and zero derivative. This also covers backbone torsions that depend on an unavailable oxygen atom.

Iterative residue frames. RFdiffusion initializes its structure-update frames as $\mathbf{R}_\ell^{(0)} = \mathbf{I}$ and $\boldsymbol{t}_\ell^{(0)} = \boldsymbol{x}_{\ell,\mathrm{C}\alpha}$, with centered coordinates $\boldsymbol{q}_{\ell a} = \boldsymbol{x}_{\ell a} - \boldsymbol{x}_{\ell,\mathrm{C}\alpha}$. These centered coordinates retain the input orientation. Each layer reconstructs global coordinates as

$$\boldsymbol{x}_{\ell a}^{(k)} = \mathbf{R}_\ell^{(k)} \boldsymbol{q}_{\ell a} + \boldsymbol{t}_\ell^{(k)}.$$

Torsion preprocessing separately constructs a local-to-global frame from N, Cα, and C; this feature frame is distinct from the structure module's identity-initialized rotation. A residue translation cancels in the centered coordinates but remains in the translation frame. The inference graph detaches rotation and translation frames in its extra-block, main-block, and refinement loops. We remove these detach operations while preserving the frame-update equations. Both translation and rotation controls can then influence later frames and their predicted coordinates.

Layerwise edge geometry. For each graph call, we retain the original selected edge indices. For an edge (ℓ, h) at layer k, we reconstruct its relative position from that layer's Cα coordinates:

$$\boldsymbol{r}_{\ell h}^{(k)} = \mathrm{sg}(\boldsymbol{r}_{\ell h,\mathrm{ref}}^{(k)}) + \left[(\boldsymbol{x}_h^{(k)} - \boldsymbol{x}_\ell^{(k)}) - \mathrm{sg}(\boldsymbol{x}_h^{(k)} - \boldsymbol{x}_\ell^{(k)})\right].$$

This restores the edge-radius dependence in SE(3) convolution. We also enable derivatives through the angular basis construction from normalized edge directions. The continuous edge geometry is recomputed at every layer; the derivative is conditional on the current discrete neighborhood branch.

Preserving forward values. Each SE(3) block uses an original inference evaluation for its reference value. A second evaluation uses the same formulas with the restored coordinate dependencies. Equation 4 combines the first value with the second Jacobian. The derivative follows the underlying real-valued geometric computation. Forward agreement and derivative accuracy are verified separately in Section 4.4. Model parameters are unchanged, and differentiation is requested only with respect to the local controls.

Conditioning and sampling state. In the evaluated self-conditioning configuration, the template coordinates and all 44 pair-template channels are determined by the previous prediction. They are held fixed when differentiating the current input. Each derivative evaluation preserves the previous prediction and the Python, NumPy, and Torch random states. The translation and rotation calls therefore use the same conditioning state. The reported implementation uses nonsymmetric Base SelfConditioning in evaluation mode, with float32 production arithmetic and no sidechain input. All internal graph calls check for finite edge lengths of at least 0.1 Å. A refused geometry check leaves that step to the native sampler without a geometric update.

B.3 Proper Kabsch alignment and the motif derivative

Let $\mathbf{Z} \in \mathbb{R}^{m\times 3}$ be the predicted motif coordinates and let $\mathbf{Y}$ be the fixed target. All N, Cα, and C atoms across all motif fragments participate in one alignment. Set $\mathbf{A} = \mathbf{C}_m\mathbf{Z}$ and $\mathbf{B} = \mathbf{C}_m\mathbf{Y}$. For $\mathbf{A}^\top\mathbf{B} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$, choose a proper Kabsch rotation

$$\mathbf{R}_* = \mathbf{U}\ \mathrm{diag}\big(1, 1, \det(\mathbf{U}\mathbf{V}^\top)\big)\ \mathbf{V}^\top.$$

The implementation computes this rotation using a float64 SVD, then holds it fixed during backpropagation. For this rotation, define

$$\ell_{\mathbf{R}_*}(\mathbf{Z}) = \left[\frac{1}{m}\|\mathbf{C}_m\mathbf{Z}\mathbf{R}_* - \mathbf{C}_m\mathbf{Y}\|_F^2 + \varepsilon\right]^{1/2}, \qquad \varepsilon = 10^{-8}\,\text{Å}^2.$$

Writing $\mathbf{E} = \mathbf{C}_m\mathbf{Z}\mathbf{R}_* - \mathbf{C}_m\mathbf{Y}$, its derivative is

$$\nabla_{\mathbf{Z}}\ell_{\mathbf{R}_*}(\mathbf{Z}) = \frac{\mathbf{C}_m\mathbf{E}\mathbf{R}_*^\top}{m\,\ell_{\mathbf{R}_*}(\mathbf{Z})}.$$

The state loss in equation 2 is $\mathcal{L}_{\mathbf{R}_*}(H) = \ell_{\mathbf{R}_*}(\mathbf{Z}(H))$. Centering remains differentiable in this expression. At differentiable configurations, the envelope theorem identifies this with the derivative of the optimized alignment loss.

For the current state H and any optimal rotation $\mathbf{R}_*$, including a nonunique optimum, define

$$\mathcal{U}_H(\boldsymbol{u}, \boldsymbol{\omega}) = \mathcal{L}_{\mathbf{R}_*}(\mathcal{T}_{\boldsymbol{u},\boldsymbol{\omega}}H).$$

Evaluating the alignment minimum at $\mathbf{R}_*$ gives

$$\mathcal{L}(\mathcal{T}_{\boldsymbol{u},\boldsymbol{\omega}}H) \le \mathcal{U}_H(\boldsymbol{u}, \boldsymbol{\omega}), \qquad \mathcal{U}_H(0,0) = \mathcal{L}(H). \tag{17}$$

B.4 Complete gated guidance policy

The implemented reward is $-\mathcal{L}$. Let $\Pi_{\mathcal{M}^c}$ zero the motif control entries. Both derivative calls start from the same unperturbed coordinates and preserve the conditioning history and random states. One call varies translation while holding rotation at zero; the other varies rotation while holding translation at zero. They therefore evaluate the two blocks of the same local objective before either update is applied. The two directions are

$$\boldsymbol{g}_u = -\Pi_{\mathcal{M}^c}\nabla_{\boldsymbol{u}}\mathcal{U}_H(0,0), \qquad \boldsymbol{g}_\omega = -\Pi_{\mathcal{M}^c}\nabla_{\boldsymbol{\omega}}\mathcal{U}_H(0,0).$$

For $\boldsymbol{v}_{\ell a} = \boldsymbol{x}_{\ell a} - \boldsymbol{x}_{\ell,\mathrm{C}\alpha}$, define the positive normalization scales

$$\nu_u = \max\left\{10^{-12}, \left(\frac{1}{L}\sum_{\ell}\|\boldsymbol{g}_{u,\ell}\|^2\right)^{1/2}\right\},$$

$$\nu_\omega = \max\left\{10^{-12}, \left(\frac{1}{3L}\sum_{\ell}\sum_{a\in\mathcal{B}}\|\boldsymbol{g}_{\omega,\ell}\times\boldsymbol{v}_{\ell a}\|^2\right)^{1/2}\right\}.$$

The rotational scale measures the induced atom-displacement tangent. Both averages include all residues, with motif controls already zero. The normalized translation and rotation directions are, respectively,

$$\boldsymbol{d}_u = \boldsymbol{g}_u/\nu_u, \qquad \boldsymbol{d}_\omega = \boldsymbol{g}_\omega/\nu_\omega.$$

The base displacement budget, in angstroms, is

$$\bar{D}_t = \begin{cases} 0, & t \ge 25, \\ 0.3(25-t)/15, & 10 < t < 25, \\ 0.3, & 1 \le t \le 10. \end{cases}$$

Let G_t be the most recent finite motif loss recorded before the current step's first schedule query. This record is the loss computed for an earlier gradient evaluation. With $\sigma(z) = (1+e^{-z})^{-1}$, the gate is

$$\gamma_t(G_t) = \begin{cases} 1, & t > 16 \text{ or no finite record exists}, \\ \sigma\big((G_t - 0.75\,\text{Å})/(0.15\,\text{Å})\big), & \text{otherwise}. \end{cases}$$

The effective budget is $D_t = \bar{D}_t\gamma_t(G_t)$. The gate reuses recorded errors, so it requires no additional model evaluation. Its value is cached within each step and follows the restored history after a parent is selected for continuation.

For a fixed state and direction, let

$$\boldsymbol{x}(\alpha) = \mathcal{T}_{\alpha \boldsymbol{d}_u, \alpha \boldsymbol{d}_\omega} \boldsymbol{x}, \qquad \Delta_x(\alpha) = \left[\frac{1}{3L} \sum_{\ell} \sum_{a \in \mathcal{B}} \|\boldsymbol{x}(\alpha)_{\ell a} - \boldsymbol{x}_{\ell a}\|^2 \right]^{1/2}.$$

The search starts with $l_0 = 0$ and

$$h_0 = \min\left\{ D_t, \frac{0.5}{\max\{\max_\ell \|\boldsymbol{d}_{\omega,\ell}\|, 10^{-12}\}} \right\}.$$

Algorithm 2 performs 20 midpoint displacement checks and returns the retained lower endpoint $\alpha_t = l_{20}$. Every retained lower endpoint is feasible, giving

$$\Delta_x(\alpha_t) \le D_t, \qquad \max_\ell \|\alpha_t \boldsymbol{d}_{\omega,\ell}\| \le 0.5.$$

The common rotation cap restricts the scalar search interval, and retaining tested lower endpoints ensures displacement feasibility regardless of the shape of $\Delta_x(\alpha)$. The search returns a tested feasible scale using coordinate-displacement checks. Local descent additionally requires the curvature bound L_H and step-size condition in equation 5; these quantities are separate from the displacement-based acceptance rule. The native denoising transition is applied after the coordinate update.

Algorithm 2 Budget-limited gated guidance before a native transition.

Require: Pre-step state H with coordinates $\boldsymbol{x}$, diffusion time t, and guidance records.
Ensure: Updated coordinates $\boldsymbol{x}'$ for the subsequent native denoising transition.

Compute and cache $D_t = \bar{D}_t \gamma_t(G_t)$ from the pre-query records as defined above.
if $D_t = 0$ **then**
    return $\boldsymbol{x}$.
end if
Evaluate $\boldsymbol{g}_u, \boldsymbol{g}_\omega$ in separate calls at the same unperturbed H.
    Preserve conditioning and random states across both calls; zero motif controls.
if a derivative geometry check refuses the update **then**
    return $\boldsymbol{x}$.
end if
Compute ν_u, ν_ω as above; set $\boldsymbol{d}_u = \boldsymbol{g}_u/\nu_u$, $\boldsymbol{d}_\omega = \boldsymbol{g}_\omega/\nu_\omega$.
Initialize $l = 0$ and $h = h_0$ using the rotation-limited bound above.
for 20 iterations **do**
    $a \leftarrow (l + h)/2$.
    if $\Delta_x(a) \le D_t$ **then**
        $l \leftarrow a$.
    else
        $h \leftarrow a$.
    end if
end for
$\alpha_t \leftarrow l$.
return $\boldsymbol{x}' = \mathcal{T}_{\alpha_t \boldsymbol{d}_u, \alpha_t \boldsymbol{d}_\omega} \boldsymbol{x}$.

B.5 Local and terminal implications

B.5.1 Local descent with the selected alignment

Fix the current state H and the same optimal alignment for both derivative blocks. Define $f_H(a) = \mathcal{U}_H(a\boldsymbol{d}_u, a\boldsymbol{d}_\omega)$. For exact block derivatives,

$$f'_H(0) = -\beta_H, \qquad \beta_H = \frac{\|\boldsymbol{g}_u\|_2^2}{\nu_u} + \frac{\|\boldsymbol{g}_\omega\|_2^2}{\nu_\omega} \ge 0.$$

Assume the declared local branch remains valid on $[0, \alpha]$, f'_H is absolutely continuous there, and $f''_H(a) \le L_H$ almost everywhere for some $L_H \ge 0$. Using equation 17 and integrating twice gives

$$\begin{aligned}\mathcal{L}(\mathcal{T}_{\alpha \boldsymbol{d}_u, \alpha \boldsymbol{d}_\omega} H) - \mathcal{L}(H) &\le f_H(\alpha) - f_H(0) \\ &= \alpha f'_H(0) + \int_0^\alpha (\alpha - a) f''_H(a)\, da \\ &\le -\alpha \beta_H + \tfrac{1}{2} L_H \alpha^2.\end{aligned}$$

Thus equation 5 gives

$$\alpha \beta_H \ge \tfrac{1}{2} L_H \alpha^2 \quad \Longrightarrow \quad \mathcal{L}(\mathcal{T}_{\alpha \boldsymbol{d}_u, \alpha \boldsymbol{d}_\omega} H) \le \mathcal{L}(H),$$

with strict descent under strict inequality. For $L_H > 0$, the condition is $0 \le \alpha \le 2\beta_H / L_H$; the undivided form also covers $L_H = 0$.

For numerical derivatives satisfying $\|\widehat{\boldsymbol{g}}_j - \boldsymbol{g}_j\|_2 \le \delta_j$, compute ν_j from $\widehat{\boldsymbol{g}}_j$ and set $\widehat{f}_H(a) = \mathcal{U}_H(a\widehat{\boldsymbol{g}}_u/\nu_u, a\widehat{\boldsymbol{g}}_\omega/\nu_\omega)$. Cauchy–Schwarz gives

$$\begin{aligned}-\widehat{f}'_H(0) &= \sum_{j \in \{u,\omega\}} \frac{\langle \boldsymbol{g}_j, \widehat{\boldsymbol{g}}_j \rangle}{\nu_j} \\ &\ge \sum_{j \in \{u,\omega\}} \frac{\|\widehat{\boldsymbol{g}}_j\|_2^2 - \delta_j \|\widehat{\boldsymbol{g}}_j\|_2}{\nu_j} = \beta_H^{\mathrm{lower}}.\end{aligned}$$

Under the same regularity and curvature conditions along this numerical direction,

$$\mathcal{L}(\mathcal{T}_{\alpha \widehat{\boldsymbol{g}}_u/\nu_u, \alpha \widehat{\boldsymbol{g}}_\omega/\nu_\omega} H) - \mathcal{L}(H) \le -\alpha \beta_H^{\mathrm{lower}} + \tfrac{1}{2} L_H \alpha^2.$$

These bounds concern the real-valued control map. The finite-difference checks below measure errors at selected states; uniform error bounds δ_j remain unspecified.

B.5.2 From local intervention to terminal utility

Complete transition state. Let P_e and Q_e, $e = 0, \dots, T-1$, be the unguided and guided Markov kernels for a single trajectory, indexed in execution order. They share an initial law ν, all other sampling rules, and the terminal sequence-design and refolding evaluation. The state H_e includes coordinates, sequence, layout, self-conditioning history, guidance records, and all stored controls used by future transitions. When EMPS is enabled, both kernels retain its conditional-innovation rule. The comparison precedes pool-wide rollout allocation.

Continuation values and exact transfer. For a bounded measurable terminal utility f, define the reference continuation values pointwise by

$$V_T = f, \qquad V_e(h) = P_e V_{e+1}(h) = \int V_{e+1}(h')\, P_e(h, dh'). \tag{18}$$

The kernels and recursion are defined on a common state space containing every state reachable under either process.

Writing $\mathbb{E}_{\mathrm{guided}}$ and $\mathbb{E}_{\mathrm{ref}}$ for the two path laws from ν, the exact terminal relation is

$$\mathbb{E}_{\mathrm{guided}} f(H_T) - \mathbb{E}_{\mathrm{ref}} f(H_T) = \sum_{e=0}^{T-1} \mathbb{E}_{\mathrm{guided}}\big[(Q_e - P_e) V_{e+1}(H_e)\big]. \tag{19}$$

Proof. The Markov property and equation 18 give

$$\mathbb{E}_{\mathrm{guided}}[V_{e+1}(H_{e+1}) - V_e(H_e) \mid H_e] = (Q_e - P_e) V_{e+1}(H_e).$$

Taking expectations and summing gives

$$\begin{aligned}\sum_{e=0}^{T-1} \mathbb{E}_{\mathrm{guided}}[(Q_e - P_e) V_{e+1}(H_e)] &= \mathbb{E}_{\mathrm{guided}}[V_T(H_T) - V_0(H_0)] \\ &= \mathbb{E}_{\mathrm{guided}} f(H_T) - \nu V_0 \\ &= \mathbb{E}_{\mathrm{guided}} f(H_T) - \mathbb{E}_{\mathrm{ref}} f(H_T).\end{aligned}$$

Terminal improvement requires the sum in equation 19 to be positive, so transferring the local motif-loss bound in Appendix B.5 to terminal utility requires information about the continuation values.

B.6 GRADIENT FIDELITY AND LOCAL-RESPONSE VALIDATION

The reference is a central finite difference of the scalar motif reward evaluated on the raw RFdiffusion output. We enumerate every free translation or rotation control coordinate and use two perturbation sizes for each state. Sequence and the matching self-conditioning history are fixed. The diagnostic reference uses float64 arithmetic and replays the layerwise edge identities recorded from the unperturbed production float32 forward. Each perturbed evaluation recomputes continuous positions and basis features from that layer's coordinates. The $t = 40$ reference uses direct float64 pairwise distances. Production distance calculations remain unchanged.

Table 8: Perturbations used for the complete-coordinate finite-difference reference. The last column compares the two resulting finite-difference vectors.

State	Control	Perturbation sizes	Dimensions	FD angle (°)
1QY3, seed 1, $t = 10$	Translation	$0.003/0.001$ Å	627	0.3854
1QY3, seed 1, $t = 10$	Rotation	$10^{-4}/(3\times10^{-5})$ rad	627	0.3098
5YUI, seed 0, $t = 40$	Translation	$10^{-6}/(3\times10^{-7})$ Å	192	0.0058

Table 8 specifies the three state–control pairs, totaling 1,446 control coordinates. For a reference vector g_{FD}, we measure the angle to the production derivative and the norm ratio $\|g\|/\|g_{\mathrm{FD}}\|$. Both vectors exclude motif controls. Forward preservation is checked independently against the native float32 raw prediction at each tested state. The float64 reference lifts the precision of the same local computation; production forward values retain native arithmetic.

Results at each perturbation size. Table 9 reports the complete step-resolved derivative checks using the protocol in Appendix B.6. For an automatic derivative g and its finite-difference reference g_{FD}, the relative error is $\|g - g_{\mathrm{FD}}\|_2/\|g_{\mathrm{FD}}\|_2$. Each row compares the fully restored production float32 derivative with the corresponding float64 finite-difference reference. The two step sizes assess local numerical agreement at each fixed state.

Table 9: Step-resolved checks of the fully restored production derivative against finite differences. Translation perturbations are in angstroms; rotation perturbations are in radians.

State / control	h	Angle (°)	Norm ratio	Relative L_2
1QY3, $t = 10$, Trans.	3×10^{-3}	0.625	0.99859	0.01098
1QY3, $t = 10$, Trans.	10^{-3}	0.376	0.99927	0.00660
1QY3, $t = 10$, Rot.	10^{-4}	0.188	1.00044	0.00331
1QY3, $t = 10$, Rot.	3×10^{-5}	0.335	1.00022	0.00585
5YUI, $t = 40$, Trans.	10^{-6}	0.475	1.00737	0.01112
5YUI, $t = 40$, Trans.	3×10^{-7}	0.471	1.00730	0.01101

The finite-difference norm ratios, with the larger-step norm in the numerator, are 1.000675, 0.999788, and 0.999929 for translation at 1QY3, rotation at 1QY3, and translation at 5YUI, respectively. Together with the reference angles in Table 8, these values quantify the change in the numerical reference as the perturbation decreases. The forward check remains separate: the maximum absolute difference from the native raw float32 coordinates is zero at both tested states.

Same-state intervention response. Eight earlier ungated trajectories across four tasks supplied 40 recorded states, including eight zero-budget controls at $t = 50$ and 32 positive-budget states. At each state, the comparison restores the same noisy state, sequence, self-conditioning history, and random state, and omits only the current guidance update. Among the 32 positive-budget probes, the raw clean motif NCAC error decreases in 31, with a median reduction of 0.0227 Å. The returned production clean NCACO error also decreases in 31 probes, with a median reduction of 0.0283 Å. These measurements quantify the immediate response of RFdiffusion's clean prediction to the restored gradient. The one increase occurs at the final step of a 6E6R trajectory, where the raw clean motif NCAC error changes from 0.381779 to 0.384139 Å. Together with the finite-difference checks, these interventions support gradient fidelity and frequent local improvement on

the examined ungated states. Applying the conditional descent bound to an individual probe would additionally require certifying its curvature and step-size assumptions. The retained records include task, root seed, diffusion time, displacement budget, and before/after errors for both raw NCAC and returned-clean NCACO predictions. These atom sets and representations are stored separately. Appendix G.5 gives the complete plotting protocol.

B.7 Evidence for state-dependent gating

Generated geometry and endpoint success. Our baseline audit examined 3,000 public RFdiffusion designs, each evaluated with eight sequences. Of the 1,419 designs with generated-motif NCACO RMSD below 0.5 Å, 364 succeed; the corresponding count is 117 of 480 designs in the interval $[0.5, 1)$ Å. None of the 1,101 designs at or above 1 Å succeeds. These bins show a strong association between generated geometry and endpoint success in the evaluated pool. They motivate correcting generated motif geometry during sampling; the control objective uses raw clean NCAC coordinates as defined in equation 2.

We then examined a three-variant pilot on tasks 04_5WN9, 05_5ZE9, 16_7BNY, 24_1QY3, and 27_4XOJ. Each task used 20 root seeds shared by guidance off, ungated guidance, and gated guidance. These were complete generated trajectories without rollout reweighting. Each final backbone received eight fresh sequence-design and refolding evaluations. Table 10 separates scaffold consistency, motif recovery, and their same-sequence joint success. Ungated guidance increased motif recovery but reduced scaffold consistency relative to guidance off. The gated variant recovered part of the scaffold-consistency count while retaining less motif correction. This pilot characterizes the competing effects used to motivate the error-dependent budget.

The gate records distinguish large and small residual motif errors. At $t = 16$, the median recorded errors for tasks 24_1QY3 and 27_4XOJ were 2.011 and 2.506 Å; their median gate weights exceeded 0.999. The other three tasks had median errors of 0.176–0.271 Å and median gate weights of 0.021–0.039. Thus the state gate retained the budget for large residuals and reduced it near the target. For tasks 24_1QY3 and 27_4XOJ, gated guidance retained approximately 95% and 98% of the ungated improvement in final generated-motif NCAC RMSD. Each percentage is the ratio of the median paired improvement relative to guidance off, computed within the task. The fixed schedule starts error-based modulation at $t = 16$, a design choice intended to avoid gating on coarse early clean predictions.

Task-level pilot and gate profiles. Table 11 gives the task-level results behind Table 10. The three variants share 20 root seeds per task and generate complete 50-step trajectories without rollout reweighting. Each backbone receives its own fresh eight-sequence evaluation stream. The five tasks were selected using effects observed in earlier runs, so this pilot describes a selected diagnostic panel.

Across the 100 designs per variant, Off, Ungated, and Gated have 66, 53, and 57 designs passing scaffold consistency, and 9, 15, and 10 passing original-target motif recovery. The corresponding same-sequence joint counts are 9, 14, and 9. For Gated versus Ungated, three paired roots succeed only under Gated and eight only under Ungated. For Gated versus Off, the corresponding counts are six and six. For Ungated versus Off, they are nine and four. These measurements separate the scaffold-consistency response from joint endpoint success.

The final columns of Table 11 use the historical loss record read at the start of the $t = 16$ step, as defined in Appendix B.4. The large residuals in 1QY3 and 4XOJ retain nearly the full budget, while the other three tasks receive much smaller weights. For 1QY3, the median paired improvement in final generated-motif NCAC error relative to Off is 1.5293 Å under Ungated and 1.4522 Å under Gated. For 4XOJ, these improvements are 2.3606 and 2.3210 Å. The ratios are 94.96% and 98.32%. Each ratio measures the fraction of the paired median geometric improvement retained by gating.

Table 10: Diagnostic pilot with 100 backbones per variant (five tasks, 20 shared root seeds per task). The two component columns count backbones with at least one of eight sequences passing the indicated criterion. Joint success requires both criteria in the same sequence.

Guidance	Scaffold < 2 Å	Motif < 1 Å	Joint success
Off	66	9	9
Ungated	53	15	14
Gated	57	10	9

Table 11: Paired five-task guidance pilot. Each variant has 20 designs per task. Success counts use eight fresh sequence evaluations per design. The final two columns summarize the gated variant's recorded input and weight at $t = 16$.

Task	Off	Ungated	Gated	Median G_{16} (Å)	Median gate weight
04_5WN9	0	0	1	0.271	0.03938
05_5ZE9	8	10	5	0.176	0.02136
16_7BNY	1	1	1	0.260	0.03683
24_1QY3	0	2	1	2.011	0.99977
27_4XOJ	0	1	1	2.506	0.99999
Total	9	14	9		

Full-benchmark comparison. Table 16 compares gated and ungated guidance in separate full-benchmark pools, both without and with rollout reweighting. Without rollout reweighting, the ungated and gated pools respectively yield Score, solved tasks, successful designs, and successful clusters of $22.46/19/483/215$ and $23.24/18/506/216$. With rollout reweighting, the corresponding observed values are $26.45/20/597/202$ and $28.83/22/684/214$. Across these independent pools, the gated configurations have higher Score, successful-design yield, and successful-cluster coverage in both backgrounds, while the ungated guidance-only pool solves one additional task. These independent pools provide aggregate endpoint evidence for the gated policy. The ungated Scores are archived 1,000-subset estimates, whereas the gated Scores are exact fixed-pool expectations.

C Rollout reweighting and downstream compatibility

C.1 Downstream compatibility quantities and failure decomposition

The downstream audit distinguishes three geometric comparisons. For design i, let d_i^{gen} be the generated motif's NCACO RMSD to the original target. For its refolded sequence j, let d_{ij}^{sc} be the whole-scaffold consistency error, d_{ij}^{ret} the refolded motif's NCACO RMSD to the generated motif, and d_{ij}^{mot} the refolded motif's NCACO RMSD to the original target. The scaffold error is the mean Euclidean Cα distance after a proper whole-chain rigid alignment. Each motif error is computed after one joint proper Kabsch alignment over the N, Cα, C, and O atoms of all declared motif fragments. The same refold supplies all three measurements for a given sequence. These atom choices differ from the N, Cα, C objective used by gated gradient guidance. In particular, d_i^{gen} is an endpoint NCACO error, whereas G_t records the raw-clean NCAC guidance loss.

For the eight endpoint sequences, define

$$A_8(i) = \min_{j=1,\dots,8} \max\left\{ \frac{d_{ij}^{\mathrm{sc}}}{2\,\text{Å}}, \frac{d_{ij}^{\mathrm{ret}}}{1\,\text{Å}} \right\}, \qquad C_8(i) = \min_{j=1,\dots,8} \max\left\{ \frac{d_{ij}^{\mathrm{sc}}}{2\,\text{Å}}, \frac{d_{ij}^{\mathrm{mot}}}{1\,\text{Å}} \right\}.$$

Each maximum uses the same sequence for its two errors. Thus A_8 tests consistency with the generated structure, whereas C_8 retains recovery of the original target. For endpoint evaluation, the scaffold reference is the final backbone, and a design succeeds exactly when $C_8 < 1$ under the saved MotifBench values. The reported errors use the evaluator's three-decimal precision, and no additional confidence threshold enters this event.

In the public RFdiffusion pool, 1,419 of 3,000 designs have $d_i^{\mathrm{gen}} < 0.5\,\text{Å}$, but only 364 of these achieve endpoint success. This establishes that accurate generated geometry alone is insufficient. A separate earlier ungated-guidance pool contains 3,000 designs and 483 endpoint successes (Table 12). It is distinct from both the public RFdiffusion pool and the final gated-guidance pool; its raw and saved endpoint values give identical design-level success flags.

Table 12: Generated geometry, same-sequence scaffold consistency and motif retention, and original-target endpoint success in the earlier ungated pool. All 3,000 designs are included. $A_8 < 1$ requires scaffold-consistency error below 2 Å and motif-retention RMSD below 1 Å in the same sequence.

Generated error (Å)	A_8	Designs	Endpoint successes	Endpoint failures
Below 1	Below 1	534	482	52
Below 1	At least 1	1,875	1	1,874
At least 1	Below 1	14	0	14
At least 1	At least 1	577	0	577
All designs		3,000	483	2,517

The 52 failures with $d_i^{\mathrm{gen}} < 1\,\text{Å}$ and $A_8(i) < 1$ pass generated-geometry, same-sequence scaffold-consistency, and motif-retention checks, yet fail original-target recovery. Conversely, one endpoint success has $A_8(i) \geq 1$, confirming that retention and target recovery are different events. For tasks 03_2CGA and 08_7AD5, all 100 generated motifs satisfy $d_i^{\mathrm{gen}} < 1\,\text{Å}$. Although 79 and 55 designs, respectively, have a sequence passing scaffold consistency, neither task has a sequence passing motif retention. Whole-scaffold consistency can therefore coexist with poor local motif recovery.

Coordinate witnesses. For each of the 52 failures, the audited sequence is the one attaining $A_8(i)$. The audit aligns the generated motif to the target and the refolded motif to the generated motif, then composes these transformations using matched NCACO atoms. In the resulting common frame, the squared composed target error equals the sum of the two squared errors and their residual cross term. The cross term is positive in 44 of 52 witnesses. In eight witnesses, the sum of the two component squared errors is below $1\,\text{Å}^2$, yet the optimally aligned target error is at least 1 Å. These failures motivate using refolded-motif error, rather than motif retention alone, in equation 6.

C.2 Parent scoring and independent continuation protocol

At the rollout boundary $\tau = 10$, the sampler saves the complete pre-step parent state $H_\tau^{(i)}$, including coordinates, sequence, layout, task conditioning, self-conditioning history, and guidance state. The step then produces the processed clean structure $\mathbf{P}_i$ used only for scoring. ProteinMPNN designs four fresh sequences for $\mathbf{P}_i$, and ESMFold refolds them. Equation 6 evaluates each refold against $\mathbf{P}_i$ through d_{ij}^{sc} and against the original target through d_{ij}^{mot}. The 2 Å and 1 Å denominators align the implementation score with the two endpoint scales; the covariance identity below applies to any finite positive parent scores.

After allocation, each offspring restores its parent's saved $H_\tau^{(i)}$, rather than the processed scoring structure $\mathbf{P}_i$, and executes steps $\tau, \ldots, 1$ with a fresh random tape. Randomness is independent across offspring and independent of scoring and allocation. Each final backbone then receives eight fresh sequence-design and refolding evaluations that are independent of the four scoring evaluations. This protocol separates evidence used to allocate computation from evidence used to report endpoint success.

C.3 Stable mixed allocation and systematic resampling

To evaluate equation 6 stably, set $m_i = \min_{1 \leq j \leq 4} r_{ij}$ and compute

$$E_i = m_i - \log\left[\frac{1}{4}\sum_{j=1}^{4}\exp\bigl(-(r_{ij} - m_i)\bigr)\right] = -\log a_i. \tag{20}$$

Let $E_{\min} = \min_i E_i$ and $v_i = \exp(-(E_i - E_{\min}))$. Then the implemented weights are

$$\pi_i = \frac{0.25}{M} + 0.75\frac{v_i}{\sum_r v_r}. \tag{21}$$

Both exponent shifts preserve the mathematical scores and weights. Weights and cumulative sums use float64. The last cumulative weight is set to one before locating systematic sample positions. Allocation requires all 100 parents, exactly four paired measurements per parent, and finite nonnegative errors and energies. Invalid inputs stop allocation; they are not replaced with an alternative score.

For cumulative weights $C_0 = 0$ and $C_i = \sum_{r=1}^{i} \pi_r$, systematic resampling draws one offset $U \sim \mathrm{Unif}[0, 1/M)$ and places points $U + k/M$, $k = 0, \ldots, M-1$. The offspring count n_i is the number of points in $[C_{i-1}, C_i)$, so $\sum_i n_i = M$ and the final candidate count is unchanged. The uniform mixture gives every parent positive probability mass, but a realized systematic-resampling draw may still assign $n_i = 0$.

C.4 Conditional success gain under score–success covariance

Condition on the complete parent pool $\mathcal{H} = (H_\tau^{(i)})_{i=1}^M$ saved before step $\tau = 10$ and its observed scores $\boldsymbol{a} = (a_i)_{i=1}^M$. Each state contains the coordinates, sequence, layout, self-conditioning history, and guidance state needed by the fixed continuation policy. Let $q_{i,+}$ denote its success probability under fresh continuation and eight-sequence final evaluation, with randomness independent of scoring and allocation. Set $\boldsymbol{q}_+ = (q_{i,+})_{i=1}^M$, $\bar{q}_+ = M^{-1}\sum_i q_{i,+}$, and $\bar{a} = M^{-1}\sum_i a_i > 0$.

Define the empirical covariance with normalization $1/M$:

$$\mathrm{Cov}_{\mathrm{emp}}(\boldsymbol{a}, \boldsymbol{q}_+) = \frac{1}{M}\sum_{i=1}^{M}(a_i - \bar{a})(q_{i,+} - \bar{q}_+).$$

Let S_+ count successful offspring among the M final designs. Then the finite-pool allocation identity is

$$\mathbb{E}\left[\frac{S_+}{M}\,\middle|\,\mathcal{H}, \boldsymbol{a}\right] - \bar{q}_+ = \frac{1-\lambda}{\bar{a}}\,\mathrm{Cov}_{\mathrm{emp}}(\boldsymbol{a}, \boldsymbol{q}_+). \tag{22}$$

The reference value $\bar{q}_+$ is the expected success fraction from one fresh continuation per parent, using the same continuation and evaluation protocol.

Proof. For the weights in equation 7, set $C_0 = 0$, $C_i = \sum_{r=1}^{i} \pi_r$, and draw $U \sim \mathrm{Unif}[0, 1/M)$ independently of $\mathcal{H}, \boldsymbol{a}$. Systematic resampling gives

$$n_i = \sum_{k=0}^{M-1} \mathbf{1}\left\{U + \frac{k}{M} \in [C_{i-1}, C_i)\right\}, \qquad \sum_i n_i = M.$$

Integrating over the offset gives

$$\begin{aligned}\mathbb{E}[n_i \mid \mathcal{H}, \boldsymbol{a}] &= M\sum_{k=0}^{M-1}\int_0^{1/M} \mathbf{1}_{[C_{i-1}, C_i)}\left(u + \frac{k}{M}\right) du \\ &= M\int_0^1 \mathbf{1}_{[C_{i-1}, C_i)}(v)\, dv = M\pi_i.\end{aligned}$$

Since $\mathbb{E}[S_+ \mid \mathcal{H}, \boldsymbol{a}, \boldsymbol{n}] = \sum_i n_i q_{i,+}$, iterated expectation yields

$$\begin{aligned}\mathbb{E}\left[\frac{S_+}{M}\,\middle|\,\mathcal{H}, \boldsymbol{a}\right] - \bar{q}_+ &= \sum_i \pi_i q_{i,+} - \bar{q}_+ \\ &= \frac{1-\lambda}{\bar{a}}\left(\frac{1}{M}\sum_i a_i q_{i,+} - \bar{a}\bar{q}_+\right) \\ &= \frac{1-\lambda}{\bar{a}}\,\mathrm{Cov}_{\mathrm{emp}}(\boldsymbol{a}, \boldsymbol{q}_+).\end{aligned}$$

This proves equations 22 and 8 without requiring independent parents or offspring counts.

For $\lambda = 0.25$, the conditional gain is positive exactly when $\mathrm{Cov}_{\mathrm{emp}}(\boldsymbol{a}, \boldsymbol{q}_+) > 0$. Here "empirical covariance" is the finite-parent average involving the latent continuation-success probabilities $\boldsymbol{q}_+$; the scores a_i determine the allocation weights π_i. Its sign at the final gated, $\tau = 10$ operating point remains unmeasured.

C.5 Full-benchmark endpoint observations

The full-benchmark ablations in Table 2 compare observed outcomes across separately generated pools. Without gated gradient guidance, rollout reweighting increases the observed number of successful designs from 481 to 626 and solved tasks from 17 to 18, while successful clusters decrease from 205 to 197. With gated gradient guidance, it increases successful designs from 506 to 684 and solved tasks from 18 to 22, while successful clusters decrease from 216 to 214. In both contexts, the observed higher yield together with lower successful-cluster coverage is consistent with rollout's yield-oriented role and motivates the separate batch-coordination role of EMPS.

D EMPS construction and implementation

D.1 Equal-probability layout strata

A layout Q is a tuple of free-segment lengths satisfying the prescribed gap bounds and total scaffold length. The native accepted-layout prior is uniform over its N_Q legal tuples. We enumerate these tuples lexicographically, placing gaps between the first and last motif fragments before terminal gaps. The order within each group follows the input contig. This ordering defines a bijective rank $R_Q \in \{0, \dots, N_Q - 1\}$. Suffix counts of bounded integer sums support rank and unrank operations without enumerating the full layout set. To unrank, traverse the gap variables in order, subtracting the number of legal suffixes for each skipped gap value. The implementation uses exact integer counts.

Augment the layout rank with an independent $K_Q \sim \mathrm{Unif}\{0, \dots, 9\}$ and define $Y_Q = 10R_Q + K_Q$. The augmented prior is uniform over $10N_Q$ integers. For layout stratum $a \in \{0, \dots, 9\}$, sample

$$\begin{gathered} J_{ab} \sim \mathrm{Unif}\{0, \dots, N_Q - 1\}, \qquad Y_{ab} = aN_Q + J_{ab}, \\ Q_{ab} = \mathrm{unrank}(\lfloor Y_{ab}/10 \rfloor)\,. \end{gathered} \tag{23}$$

Each stratum has N_Q augmented atoms and probability $1/10$. Uniform mixing recovers the native prior because each layout rank has ten preimages, possibly in different strata.

Each cell (a, b) has its own independent layout draw and Gaussian-quantile draw. Here $b \in \{0, \dots, 9\}$ indexes the process-coordinate decile constructed below. The ten process cells within a layout stratum use separate layout draws. An independent uniform permutation assigns the 100 cell descriptors to the 100 parent slots. The plan is generated before sampling and binds each parent to its layout, process stratum, and scalar-noise stream.

D.2 Geometry-dependent process direction and degenerate fallback

For the free Cα matrix $\mathbf{X} \in \mathbb{R}^{r \times 3}$ and $\mathbf{C}_r$ from equation 9, define $R(\mathbf{X}) = \|\mathbf{C}_r\mathbf{X}\|_F$. At $R(\mathbf{X}) > 0$, its differential along $\mathbf{Z}$ is

$$DR(\mathbf{X})[\mathbf{Z}] = \frac{\langle \mathbf{C}_r\mathbf{X}, \mathbf{C}_r\mathbf{Z} \rangle_F}{\|\mathbf{C}_r\mathbf{X}\|_F} = \boldsymbol{v}(\mathbf{X})^\top \mathrm{vec}(\mathbf{Z}).$$

Thus $\boldsymbol{v}(\mathbf{X}) = \mathrm{vec}(\nabla R(\mathbf{X}))$ has unit Euclidean norm and reshapes to an array with zero centroid. For any translation $\mathbf{1}\boldsymbol{h}^\top$ and any skew-symmetric matrix $\boldsymbol{\Omega}$,

$$\boldsymbol{v}(\mathbf{X})^\top \mathrm{vec}(\mathbf{1}\boldsymbol{h}^\top) = 0, \qquad \boldsymbol{v}(\mathbf{X})^\top \mathrm{vec}(\mathbf{X}\boldsymbol{\Omega}) = \frac{\mathrm{tr}(\mathbf{X}^\top\mathbf{C}_r\mathbf{X}\boldsymbol{\Omega})}{\|\mathbf{C}_r\mathbf{X}\|_F} = 0. \tag{24}$$

The second equality uses symmetry of $\mathbf{X}^\top\mathbf{C}_r\mathbf{X}$ and skew-symmetry of $\boldsymbol{\Omega}$. The signed innovation is applied around the native reverse-step mean.

The implementation computes the direction from the actual clean prediction passed to the translational sampler, after the native clean-structure processing and any enabled geometric guidance. It uses float64 centering and normalization. The positive scalar coordinate scaling in the native sampler cancels in the normalized direction. When the centered norm is at most 10^{-12} in these scaled coordinates, an independent auxiliary Gaussian matrix supplies a fallback direction. It is centered when $r > 1$ and then normalized; for $r = 1$, only normalization is used. The fallback retains the unit-norm condition required by the noise construction. Equation 24 describes the nondegenerate radial branch. Motif entries are zero when the free-residue direction is embedded into the full coordinate array.

D.3 Gaussian whole-window process coordinate

Under the matched reference sampler, the causal unit directions in equation 9 make the projection vector

$$\boldsymbol{\eta}_{\mathrm{ref}} = \left(\boldsymbol{v}_{t_{\mathrm{g}}}^{\top}\boldsymbol{\epsilon}_{t_{\mathrm{g}}}, \ldots, \boldsymbol{v}_{\tau+1}^{\top}\boldsymbol{\epsilon}_{\tau+1}\right)^{\top}$$

standard Gaussian. Consequently, $S = \boldsymbol{w}^{\top}\boldsymbol{\eta}_{\mathrm{ref}} \sim \mathcal{N}(0,1)$ because $\|\boldsymbol{w}\|_2 = 1$. EMPS stratifies the aggregate signed projection over the controlled window; individual step projections follow the conditional Gaussian law below, and the full denoising trajectory determines the terminal scaffold radius. The layout and process axes therefore admit ten equal-probability strata each. Appendix E.2 gives the complete filtration argument, independence conditions, and treatment of failed paths.

D.4 Conditional innovation construction

Order the $t_{\mathrm{g}} - \tau$ components of $\boldsymbol{w}$ and $\boldsymbol{\eta}$ by denoising time $t_{\mathrm{g}}, t_{\mathrm{g}} - 1, \ldots, \tau + 1$. Since $\|\boldsymbol{w}\|_2 = 1$, equation 10 gives

$$\boldsymbol{\eta} = (\mathbf{I}_{t_{\mathrm{g}}-\tau} - \boldsymbol{w}\boldsymbol{w}^{\top})\boldsymbol{g} + \boldsymbol{w}s, \qquad \boldsymbol{\eta} \mid s \sim \mathcal{N}(\boldsymbol{w}s, \mathbf{I}_{t_{\mathrm{g}}-\tau} - \boldsymbol{w}\boldsymbol{w}^{\top}), \qquad \boldsymbol{w}^{\top}\boldsymbol{\eta} = s.$$

This is the conditional law of a standard Gaussian vector given its projection onto $\boldsymbol{w}$. The scalar vector is sampled once per trajectory. At each controlled step, its corresponding component replaces the current spatial projection onto $\boldsymbol{v}_t$. Conditional on the current history and direction, the residual spatial noise is standard Gaussian on the orthogonal complement of $\boldsymbol{v}_t$ and independent of η_t. The direction uses the available sampling history and, if needed, independent fallback randomness; it does not directly inspect the cell label, target s, or unconsumed innovations.

The first operation is the Gaussian conditional construction for the whole-window projection, and the second replaces only the current spatial projection. Mixing the scalar target over its unstratified standard-normal law recovers the native translational innovation law. Appendices E.2 and E.3 prove the exact conditional path law and random-slot marginal preservation.

D.5 Native RFdiffusion sampler interface

The layout wrapper first executes the native contig construction and propagates any initialization failure. On success, it rebuilds the residue mapping and motif mask for the assigned legal layout. Gaussian initialization then follows the native routine. The process hook acts only on the translational noise draw at $t = 25, \ldots, 11$. The native translational mean, scalar noise standard deviation, rotational sampling, and motif handling retain their original rules. In particular, the normalized innovation in equation 10 is converted back to coordinates using the native standard deviation.

For each controlled step, the implementation first makes the original normal draw. Using the coordinate arrays of Algorithm 4, the free-residue draw satisfies $\mathrm{vec}(\mathbf{A}_{t,\mathrm{free}}) = \mathrm{vec}(\mathbf{B}_{t,\mathrm{free}}) + \sigma_t\boldsymbol{\xi}_t$. It recovers the standardized innovation in float64, applies the projection, and casts the result back to the production dtype. Only free-residue entries are replaced. This retains the shape and random-number consumption of the native noise call. Outside the window, the hook directly returns the native draw.

D.6 Interaction with guidance and the rollout boundary

Within each controlled denoising step, gated gradient guidance first updates the current state. The production RFdiffusion forward pass then returns the processed clean prediction used to form $\boldsymbol{v}_t$,

after which EMPS replaces only the translational innovation. Rotational sampling and the remaining native state updates are unchanged. Thus the process direction reflects the state produced by any enabled guidance while the conditional-noise construction retains the native transition law after mixing over cells.

The experimental EMPS window is $t = 25, \ldots, 11$. It ends immediately before the complete parent state $H_\tau^{(i)}$ is saved at $\tau = 10$. An offspring selected by rollout reweighting restores this state, retains its parent's layout and motif mapping, and executes $t = 10, \ldots, 1$ with fresh native noise and the EMPS process hook inactive. The RFdiffusion-with-EMPS and single-axis ablations run uninterrupted through all 50 denoising steps. Under the independent-random-tape assumptions of Appendix E.1, one fresh continuation from the saved complete state has the same conditional tail law as uninterrupted generation.

These conditional-path and random-slot marginal results describe the matched-reference parent pool before rollout allocation. Appendix F analyzes the resulting offspring pool, including the coverage effects of parents that are omitted or assigned additional independent continuations.

D.7 EMPS PSEUDOCODE

Algorithms 3 and 4 specify Exploration via Marginal-Preserving Stratification (EMPS). The first constructs the batch plan. The second replaces the translational innovation at its native sampling site. Appendix E.2 proves the conditional-path and random-slot marginal properties, and Appendix E analyzes finite-budget coverage under the matched reference. We use vec to flatten coordinate arrays and unvec to restore their shape. The frozen configuration uses 100 parent slots, ten strata per coordinate, and EMPS steps $t = 25, 24, \ldots, 11$.

Plan storage and initialization. Each cell has separate layout, quantile, and scalar-Gaussian streams. The assignment permutation uses a fourth stream. The implementation derives the cell streams deterministically from the master seed, task identifier, cell indices, and stream role. The permutation stream uses the master seed, task identifier, and its own role. It stores the scalar-Gaussian seed in the descriptor and reconstructs $\boldsymbol{\eta}$ once when that parent's process hook is activated. This is the same construction as Algorithm 3. The descriptor remains bound to the parent when tasks are scheduled or restarted.

For initialization, first call the native contig constructor. On success, rebuild its residue mapping and motif mask with the assigned legal tuple Q_i. Then complete the native Gaussian initialization. The assigned tuple is expressed in the original contig order. This binding is also used when reconstructing the sampler for a saved parent's continuation.

Implementation notes. The component ordering and rollout boundary are specified in Appendix D.6. The native sampler supplies Algorithm 4 with its actual clean prediction, reverse-step mean, and noise scale. The centered-norm threshold is measured in the native sampler's scaled coordinates. The positive scalar coordinate scaling cancels in the normalized direction. Direction computation and projection use float64; only the replacement coordinates are cast back. The auxiliary direction stream is separate from the scalar innovations and native noise. It is indexed by the task, parent seed, parent slot, and stream role.

Only free-residue entries of the translational draw are replaced. After parent generation, require that the record contains exactly the 15 prescribed steps in order. An invalid endpoint, nonfinite quantity, or failed check terminates the run without a replacement draw. The two algorithms use no additional model evaluations.

D.8 NUMERICAL AND EXECUTION CHECKS

Independent private streams generate the refined layout draws, process-coordinate targets, scalar Gaussian vectors, assignment permutation, and auxiliary fallback directions. Runtime guards verify valid Gaussian-CDF endpoints, the prescribed 15-step order, finite values, and realization of every scalar projection within an absolute tolerance of 2×10^{-4}. A failed guard terminates the run without a replacement draw. The probability-law results in Appendix E.2 concern the ideal real-valued construction; the checks below audit its finite-precision execution.

Algorithm 3 EMPS batch plan and whole-window scalar innovations.

Require: Motif contig, free-segment bounds, prescribed total length; 100 parent slots with distinct native seeds; private master seed.

Ensure: One descriptor $(Q_i, s_i, \boldsymbol{\eta}_i)$ for each parent slot i.

Order free segments: internal linkers first, then terminal segments; retain contig order within each group.
Build exact suffix counts for legal gap tuples in this order; require their number $N_Q > 0$.
Set $\boldsymbol{w} = \mathbf{1}/\sqrt{15}$, indexed in denoising order $t = 25, 24, \ldots, 11$.
for $a = 0, \ldots, 9$ **do**
    for $b = 0, \ldots, 9$ **do**
        Draw J_{ab} uniformly from $\{0, \ldots, N_Q - 1\}$.
        Set $Y_{ab} = aN_Q + J_{ab}$ and $R_{ab} = \lfloor Y_{ab}/10 \rfloor$.
        Unrank R_{ab} using the suffix counts; restore the gaps to their original contig order to obtain Q_{ab}.
        Draw $V_{ab} \sim \mathrm{Unif}(0, 1)$ independently; set $U_{ab} = (b + V_{ab})/10$.
        Require $0 < U_{ab} < 1$ and $b/10 \le U_{ab} < (b + 1)/10$; set $s_{ab} = \Phi^{-1}(U_{ab})$.
        Draw $\boldsymbol{g}_{ab} \sim \mathcal{N}(0, \mathbf{I}_{15})$ independently.
        Set $\boldsymbol{\eta}_{ab} = \boldsymbol{g}_{ab} + \boldsymbol{w}(s_{ab} - \boldsymbol{w}^\top \boldsymbol{g}_{ab})$.
        Require $|\boldsymbol{w}^\top \boldsymbol{\eta}_{ab} - s_{ab}| \le 10^{-12}$.
        Store $d_{ab} = (Q_{ab}, s_{ab}, \boldsymbol{\eta}_{ab})$.
    end for
end for
Uniformly permute the 100 descriptors, independently of all cell draws.
Bind the permuted descriptors to the parent slots and native seeds.
return the bound descriptors.

Algorithm 4 EMPS replacement of one native translational noise draw.

Require: Step t; native mean $\mathbf{B}_t$ and standard deviation σ_t; all-residue production clean Cα coordinates $\mathbf{X}^{\mathrm{clean}}_{t,\mathrm{all}} \in \mathbb{R}^{L\times 3}$; motif mask; parent slot i's descriptor $(Q_i, s_i, \boldsymbol{\eta}_i)$ and consumed-step record, if active.

Ensure: A translational draw $\mathbf{A}_t$ with the native shape and dtype.

if the process hook is inactive or $t < 11$ or $t > 25$ **then**
    return one native draw $\mathrm{vec}(\mathbf{A}_t) \sim \mathcal{N}(\mathrm{vec}(\mathbf{B}_t), \sigma_t^2 \mathbf{I}_{3L})$.
end if
Require t to be the next unconsumed member of $(25, 24, \ldots, 11)$.
Require finite, positive scalar σ_t and at least one free residue.
Extract the free-residue coordinates $\mathbf{X}^{\mathrm{clean}}_t \in \mathbb{R}^{r\times 3}$ from $\mathbf{X}^{\mathrm{clean}}_{t,\mathrm{all}}$ in float64.
Set $\mathbf{Z} = \mathbf{X}^{\mathrm{clean}}_t - \mathbf{1}\,\mathrm{mean}(\mathbf{X}^{\mathrm{clean}}_t)$; require finite $\mathbf{Z}$.
if $\|\mathbf{Z}\|_F \le 10^{-12}$ **then**
    Draw $\mathbf{Z} \in \mathbb{R}^{r\times 3}$ with independent standard Gaussian entries from the auxiliary direction stream.
    if $r > 1$ **then**
        Set $\mathbf{Z} \leftarrow \mathbf{Z} - \mathbf{1}\,\mathrm{mean}(\mathbf{Z})$.
    end if
end if
Require finite, positive $\|\mathbf{Z}\|_F$; set $\boldsymbol{v}_t = \mathrm{vec}(\mathbf{Z})/\|\mathbf{Z}\|_F$.
Draw the full native array $\mathrm{vec}(\mathbf{A}_t) \sim \mathcal{N}(\mathrm{vec}(\mathbf{B}_t), \sigma_t^2 \mathbf{I}_{3L})$ once.
Recover $\boldsymbol{\xi}_t = \mathrm{vec}(\mathbf{A}_{t,\mathrm{free}} - \mathbf{B}_{t,\mathrm{free}})/\sigma_t$ in float64.
Set $\boldsymbol{\epsilon}_t = \boldsymbol{\xi}_t + \boldsymbol{v}_t(\eta_{i,t} - \boldsymbol{v}_t^\top \boldsymbol{\xi}_t)$.
Replace $\mathbf{A}_{t,\mathrm{free}} \leftarrow \mathbf{B}_{t,\mathrm{free}} + \sigma_t\,\mathrm{unvec}(\boldsymbol{\epsilon}_t)$, cast to the native dtype.
Recompute $\widehat{\eta}_t = \boldsymbol{v}_t^\top\,\mathrm{vec}(\mathbf{A}_{t,\mathrm{free}} - \mathbf{B}_{t,\mathrm{free}})/\sigma_t$ in float64.
Require finite $\widehat{\eta}_t$ and $|\widehat{\eta}_t - \eta_{i,t}| \le 2 \times 10^{-4}$.
Record the consumed step and realized projection; **return** $\mathbf{A}_t$.

The numerical checks were audited on the four original axis-ablation tasks: 06_6E6R, 11_3TQB, 14_5IUS, and 15_7A8S. Joint EMPS, Layout-only, and Process-only each have 100 completed designs per task. The resulting 1,200 trajectories contain 18,000 controlled translational steps, all in the prescribed $25, \ldots, 11$ order. Every trajectory executes 50 production forwards and two restored warmup forwards. The recorded number of additional EMPS model calls is zero.

The largest absolute error in the realized weighted process coordinate is 5.77×10^{-7}. The largest per-step scalar projection error is 9.51×10^{-7}, below the implemented 2×10^{-4} tolerance. No fallback direction was used in these trajectories. The audit also checks slot completeness, the active layout and process strata, and the retained generation and endpoint seed pairings.

In Layout-only, the layout strata are retained and the scalar target has its unstratified standard-normal law. In Process-only, the process strata are retained and the layout rank follows the unstratified legal-layout prior. Both variants use the same conditional-noise implementation as Joint EMPS. Integrating over the unstratified scalar target in Layout-only recovers the reference process law. These execution checks audit the implementation of the sampling construction.

E MATCHED-REFERENCE COVERAGE THEORY

We analyze matched-reference sampling before rollout reweighting, assigning one fresh continuation to every parent and using an outputwise category map fixed before sampling. Appendix F treats rollout allocation and MotifBench's set-dependent Foldseek readout.

E.1 REFERENCE GENERATOR AND COMPARISON SCHEMES

Fix a task and a measurable single-trajectory generator $\mathcal{G}$, with path law $\mathbb{P}_{\mathrm{ref}}$ as in Section 3.6. It includes the accepted-layout prior, initialization, denoising under a fixed guidance policy, one fresh continuation per pre-$t = 10$ parent, and final evaluation with eight fresh sequences. The complete state contains coordinates, sequence, layout, task conditioning, self-conditioning history, guidance records, and every persistent control used by later transitions. Because the EMPS window ends at $t = 11$, continuing once from the complete state saved before $t = 10$ has the same conditional tail law as uninterrupted generation, excluding the unused random tape. The comparison concerns this shared path and evaluation law and excludes controller-only descriptors and auxiliary buffers from its observables.

Assume ideal real arithmetic, independent random tapes across cells and continuations, and an almost surely defined reference generator. All noncontrolled randomness retains its reference law. The law includes all paths without conditioning on completion, success, or numerical acceptance. Early failures may receive output 0, with unused auxiliary innovations keeping the process coordinate defined; if initialization can fail, an auxiliary layout preserves the declared prior. Appendices D.5 and D.8 separately document the finite-precision implementation and its numerical checks.

Let $O = \mathcal{G}(U)$ be the evaluated output on random tape U. Fix before sampling a measurable category map $\kappa(O) \in \{0, 1, 2, \ldots\}$, where label 0 denotes failure and positive labels denote successful structural categories. For a batch of $M = 100$ outputs, define

$$N_c = \sum_{i=1}^{M} \mathbf{1}\{\kappa(O_i) = c\}, \qquad S_+ = \sum_{c>0} N_c, \qquad K = \sum_{c>0} \mathbf{1}\{N_c > 0\}, \qquad D = \sum_{c>0} \binom{N_c}{2}.$$

Here S_+ counts successes, K counts represented successful categories, and D counts repeated successful-category pairs.

Write $\mathcal{C}_{ab}$ for layout stratum a and process stratum b, with $a, b \in \{0, \ldots, 9\}$. We compare four schemes using the same $\mathcal{G}$, prior, continuation, evaluation, and candidate budget. Joint EMPS, denoted J, draws one independent conditional trajectory from every joint cell $\mathcal{C}_{ab}$. Layout-only, denoted L, draws ten independent reference trajectories within each layout stratum, with independent unstratified process coordinates. Process-only, denoted P, draws ten independent reference trajectories within each process stratum, with independent layouts from the unstratified native prior. IID sampling, denoted I, draws 100 independent reference trajectories. These symbols correspond to the schemes in equation 11.

E.2 EXACT CONDITIONAL PATH LAW

Let $\mathcal{F}_{t^-}$ be generated by the realized sampler history and independent fallback-direction randomness immediately before the current translational innovation. It includes the layout and clean prediction, but excludes the controller's cell descriptor, target s, unconsumed scalar innovations, and current spatial draw. The unit vector $\boldsymbol{v}_t$ is $\mathcal{F}_{t^-}$-measurable. Flattening the free-residue coordinates, the reference innovation satisfies $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{3r})$ independently of $\mathcal{F}_{t^-}$. For $\eta_t = \boldsymbol{v}_t^\top \boldsymbol{\epsilon}_t$ and every $u \in \mathbb{R}$,

$$\mathbb{E}\big[e^{\mathrm{i}u\eta_t} \mid \mathcal{F}_{t^-}\big] = \exp(-u^2/2).$$

Iterated conditioning gives $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{t_g-\tau})$, independently of the augmented layout and pre-window history. Hence $S = \boldsymbol{w}^\top \boldsymbol{\eta} \sim \mathcal{N}(0,1)$ because $\|\boldsymbol{w}\|_2 = 1$.

Equation 23 gives N_Q augmented atoms per layout stratum and ten preimages per native layout rank, so

$$\Pr(\text{layout stratum } a) = \frac{N_Q}{10N_Q} = \frac{1}{10}, \qquad \Pr(R_Q = j) = \frac{10}{10N_Q} = \frac{1}{N_Q}, \qquad \Pr(\mathcal{C}_{ab}) = \frac{1}{100}.$$

The last equality uses independence of S and the augmented layout.

Mix the target in equation 10 over $s \sim \mathcal{N}(0,1)$, independently of $\boldsymbol{g}$ and all other random sources. Then

$$\begin{aligned} \boldsymbol{\eta} &= (\mathbf{I}_{t_g-\tau} - \boldsymbol{w}\boldsymbol{w}^\top)\boldsymbol{g} + \boldsymbol{w}s, & \boldsymbol{w}^\top\boldsymbol{\eta} &= s, \\ \mathrm{Cov}(\boldsymbol{\eta}) &= (\mathbf{I}_{t_g-\tau} - \boldsymbol{w}\boldsymbol{w}^\top)^2 + \boldsymbol{w}\boldsymbol{w}^\top = \mathbf{I}_{t_g-\tau}, & \boldsymbol{\eta} &\sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{t_g-\tau}). \end{aligned}$$

Reconstructing the path in native order gives

$$\boldsymbol{\epsilon}_t = (\mathbf{I}_{3r} - \boldsymbol{v}_t\boldsymbol{v}_t^\top)\boldsymbol{\xi}_t + \boldsymbol{v}_t\eta_t, \qquad \boldsymbol{\epsilon}_t \mid \mathcal{F}_{t^-} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{3r}).$$

Conditional on the augmented layout, induction over transitions therefore recovers the reference path law, including evaluation, with $S = s$.

The constructed pair (path, s) has the reference law of (path, S). Disintegration on standard Borel state spaces identifies the conditional path law for almost every s. Restricting s to its Gaussian quantile interval and the augmented layout to stratum a yields

$$\mathrm{Law}(\text{path in cell } \mathcal{C}_{ab}) = \mathbb{P}_{\mathrm{ref}}(\,\cdot \mid \mathcal{C}_{ab}).$$

Conditional on s, the scalar covariance is $\mathbf{I}_{t_g-\tau} - \boldsymbol{w}\boldsymbol{w}^\top$; scalar independence is used only under the mixed reference law.

E.3 RANDOM-SLOT MARGINAL PRESERVATION

Let Π be the independent uniform permutation of the 100 cells into returned slots. For any measurable path event E and any slot i,

$$\Pr_{\mathrm{J}}(\mathrm{path}_i \in E) = \frac{1}{100}\sum_{a,b} \mathbb{P}_{\mathrm{ref}}(E \mid \mathcal{C}_{ab}) = \mathbb{P}_{\mathrm{ref}}(E).$$

Together with Appendix E.2, this proves Theorem 1. Every returned slot therefore has the full reference marginal on sampling and evaluation paths. The slots are not independent after permutation; their coordinated dependence is the stratification mechanism. Permutation-invariant batch statistics may nevertheless be analyzed using the independent trajectories in cell order.

E.4 EXPECTED-SUCCESS CONSERVATION

For each successful category $c > 0$, define

$$p_{ab,c} = \Pr\{\kappa(O) = c \mid \mathcal{C}_{ab}\}, \qquad \bar{p}_{a,c} = \frac{1}{10}\sum_b p_{ab,c},$$

$$\widetilde{p}_{b,c} = \frac{1}{10}\sum_a p_{ab,c}, \qquad p_c = \frac{1}{100}\sum_{a,b} p_{ab,c}.$$

Linearity and the common slot marginals give

$$\mathbb{E}S_{+,\mathrm{J}} = \mathbb{E}S_{+,\mathrm{L}} = \mathbb{E}S_{+,\mathrm{P}} = \mathbb{E}S_{+,\mathrm{I}} = 100 \sum_{c>0} p_c. \tag{25}$$

This is an equality of expectations over newly generated matched-reference batches.

E.5 Joint, single-axis, and IID coverage ordering

For category c, the full-batch omission probabilities under Joint, Layout-only, Process-only, and IID sampling are

$$B_c = \prod_{a,b}(1-p_{ab,c}), \qquad A_c^{\mathrm{L}} = \prod_{a}(1-\bar{p}_{a,c})^{10}, \qquad A_c^{\mathrm{P}} = \prod_{b}(1-\widetilde{p}_{b,c})^{10}, \qquad A_c^{\mathrm{I}} = (1-p_c)^{100}.$$

AM–GM within layout rows or process columns, followed by AM–GM across all slots, yields

$$B_c \le A_c^{\mathrm{L}} \le A_c^{\mathrm{I}}, \qquad B_c \le A_c^{\mathrm{P}} \le A_c^{\mathrm{I}}.$$

Summing the complementary category probabilities proves

$$\mathbb{E}K_{\mathrm{J}} \ge \mathbb{E}K_{\mathrm{L}} \ge \mathbb{E}K_{\mathrm{I}}, \qquad \mathbb{E}K_{\mathrm{J}} \ge \mathbb{E}K_{\mathrm{P}} \ge \mathbb{E}K_{\mathrm{I}}.$$

Together with equation 25, these inequalities prove Theorem 2. There is no general ordering between Layout-only and Process-only.

E.6 Strict-gain and equality conditions

Define the within-layout and within-process category variations

$$V_c^{\mathrm{L}} = \sum_{a,b}(p_{ab,c} - \bar{p}_{a,c})^2, \qquad V_c^{\mathrm{P}} = \sum_{a,b}(p_{ab,c} - \widetilde{p}_{b,c})^2.$$

For $p_{ab,c} < 1$, the function $f(p) = \log(1-p)$ satisfies $f''(p) = -(1-p)^{-2} \le -1$. Because the deviations sum to zero within each row or column, its quadratic upper bound gives

$$\begin{aligned} \mathbb{E}K_{\mathrm{J}} - \mathbb{E}K_{\mathrm{L}} &\ge \sum_{c>0} A_c^{\mathrm{L}}\big(1 - e^{-V_c^{\mathrm{L}}/2}\big), \\ \mathbb{E}K_{\mathrm{J}} - \mathbb{E}K_{\mathrm{P}} &\ge \sum_{c>0} A_c^{\mathrm{P}}\big(1 - e^{-V_c^{\mathrm{P}}/2}\big). \end{aligned} \tag{26}$$

If some $p_{ab,c} = 1$, the corresponding Joint omission probability is zero and the same bounds remain valid, including when the single-axis omission probability is also zero.

Joint EMPS gains strictly over Layout-only whenever $A_c^{\mathrm{L}} V_c^{\mathrm{L}} > 0$ for at least one successful category, and the analogous condition $A_c^{\mathrm{P}} V_c^{\mathrm{P}} > 0$ gives a strict gain over Process-only. The two factors have distinct meanings: the added axis must distinguish the category, and the comparison scheme must still have a positive probability of missing it. If every $V_c^{\mathrm{L}} = 0$, Joint and Layout-only have identical categorical laws in cell order; the analogous statement holds for Process-only. If category probabilities vary only with layout, Joint and Layout-only coincide while Process-only has the IID categorical law; exchanging the axes reverses these roles. Evaluating these strictness conditions requires information about $p_{ab,c}$ beyond the single category label observed in each cell.

E.7 Repeated categories and category-count variance

For independent categorical outputs with slot probabilities $p_{i,c}$,

$$\mathbb{E}\binom{N_c}{2} = \frac{1}{2}\left[\left(\sum_i p_{i,c}\right)^2 - \sum_i p_{i,c}^2\right], \qquad \mathrm{Var}(N_c) = \sum_i p_{i,c} - \sum_i p_{i,c}^2.$$

The category means agree across the matched schemes, and $\sum_{a,b} p_{ab,c}^2 - 10\sum_a \bar{p}_{a,c}^2 = V_c^{\mathrm{L}}$. Therefore

$$\mathbb{E}D_{\mathrm{L}} - \mathbb{E}D_{\mathrm{J}} = \frac{1}{2}\sum_{c>0} V_c^{\mathrm{L}}, \qquad \mathrm{Var}_{\mathrm{L}}(N_c) - \mathrm{Var}_{\mathrm{J}}(N_c) = V_c^{\mathrm{L}}. \tag{27}$$

The columnwise analogues replace L and V_c^{L} with P and V_c^{P}.

Variance of general observables. Let $f(O)$ be square integrable and set $m_{ab} = \mathbb{E}[f(O) \mid \mathcal{C}_{ab}]$ and $m_a = 10^{-1}\sum_b m_{ab}$. For $\bar{f} = M^{-1}\sum_i f(O_i)$, independence and total variance give

$$\begin{aligned}\mathrm{Var_L}(\bar{f}) - \mathrm{Var_J}(\bar{f}) &= \frac{1}{M^2}\sum_a \left[10\,\mathrm{Var}(f(O) \mid \text{layout stratum } a) - \sum_b \mathrm{Var}(f(O) \mid \mathcal{C}_{ab})\right] \\ &= \frac{1}{M^2}\sum_{a,b}(m_{ab} - m_a)^2.\end{aligned}$$

For $f = \mathbf{1}\{\kappa(O) > 0\}$, this variance gain can vanish even when the category probabilities differ within a row.

E.8 Uniform subset coverage

Select k of the $M = 100$ returned slots uniformly without replacement, independently of generation. Let $K_{r,k}$ and $D_{r,k}$ be the successful-category coverage and repeated-category pairs in this subset under scheme r. For one category, write $q_i = 1 - p_{i,c}$ and let $e_k(\boldsymbol{q}) = \sum_{|T|=k}\prod_{i\in T} q_i$, with $e_0 = 1$. Independence in cell order gives

$$\Pr(N_{c,k} = 0) = \frac{e_k(q_1, \ldots, q_M)}{\binom{M}{k}}.$$

For $k \geq 2$, let $\boldsymbol{q}_{\mathrm{rest}}$ collect all coordinates except i, j and set $\bar{q} = (q_i + q_j)/2$. Then

$$e_k(\boldsymbol{q}) = e_k(\boldsymbol{q}_{\mathrm{rest}}) + (q_i + q_j)e_{k-1}(\boldsymbol{q}_{\mathrm{rest}}) + q_i q_j e_{k-2}(\boldsymbol{q}_{\mathrm{rest}}),$$

$$e_k(\boldsymbol{q}_{\mathrm{rest}}, \bar{q}, \bar{q}) - e_k(\boldsymbol{q}) = \frac{(q_i - q_j)^2}{4}e_{k-2}(\boldsymbol{q}_{\mathrm{rest}}) \geq 0.$$

Repeated pair averaging within rows or columns and then across all slots yields

$$\mathbb{E}K_{\mathrm{J},k} \geq \mathbb{E}K_{\mathrm{L},k} \geq \mathbb{E}K_{\mathrm{I},k}, \qquad \mathbb{E}K_{\mathrm{J},k} \geq \mathbb{E}K_{\mathrm{P},k} \geq \mathbb{E}K_{\mathrm{I},k},$$

with equality for $k = 0, 1$.

Each pair of slots is selected with probability $k(k-1)/(M(M-1))$, so equation 27 gives

$$\mathbb{E}D_{\mathrm{L},k} - \mathbb{E}D_{\mathrm{J},k} = \frac{k(k-1)}{M(M-1)}\,\frac{1}{2}\sum_{c>0} V_c^{\mathrm{L}},$$

with the analogous Process-only identity. These expectations integrate both newly generated matched-reference batches and random subset selection. Figure 3 reports coverage conditional on a saved generated pool and its full-pool cluster labels, averaging over uniform subset selection.

E.9 Concave coverage utilities

For every nondecreasing, discretely concave $\phi : \{0, \ldots, k\} \to \mathbb{R}$,

$$\begin{aligned}\mathbb{E}\phi(K_{\mathrm{J},k}) \geq \mathbb{E}\phi(K_{\mathrm{L},k}) \geq \mathbb{E}\phi(K_{\mathrm{I},k}),\\ \mathbb{E}\phi(K_{\mathrm{J},k}) \geq \mathbb{E}\phi(K_{\mathrm{P},k}) \geq \mathbb{E}\phi(K_{\mathrm{I},k}).\end{aligned} \tag{28}$$

Categorical pair averaging. For an independent categorical array, replacing two probability vectors by their mean cannot increase the expected utility of subset coverage under uniform independent selection.

Proof. First take finitely many successful categories. Write the two probability vectors as $\boldsymbol{m} + \boldsymbol{d}$ and $\boldsymbol{m} - \boldsymbol{d}$. Condition on the selected subset of all other slots and their outputs. Let h be their covered-category count and $\mathcal{U}$ their set of uncovered successful categories. If both averaged slots are selected, let ΔK be their number of new categories and define $\delta_1 = \phi(h+1) - \phi(h)$ and $\delta_2 = \phi(h+2) - \phi(h+1)$, so $\delta_1 \geq \delta_2 \geq 0$. Set $x = \sum_{c\in\mathcal{U}} m_c$ and $y = \sum_{c\in\mathcal{U}} d_c$. The unequal laws give

$$\begin{aligned}\Pr(\Delta K \geq 1) &= 1 - (1-x)^2 + y^2,\\ \Pr(\Delta K = 2) &= x^2 - y^2 - \sum_{c\in\mathcal{U}} m_c^2 + \sum_{c\in\mathcal{U}} d_c^2.\end{aligned}$$

Subtracting the utility under the averaged laws gives

$$(\delta_1 - \delta_2)\left(\sum_{c\in\mathcal{U}} d_c\right)^2 + \delta_2 \sum_{c\in\mathcal{U}} d_c^2 \geq 0. \tag{29}$$

If exactly one slot is selected, either index is equally likely and has the same mean law $\boldsymbol{m}$; if neither is selected, the utilities coincide. For countably many categories, merge labels above L into failure and let $L \to \infty$. Since coverage is bounded by k, bounded convergence extends the inequality. □

Cycling through pairs decreases the squared distance to the group mean and converges to equal probability vectors. Applying this operation within layout rows or process columns and then across all slots proves equation 28. Expected utility is a continuous polynomial for finitely many labels; the same truncation handles countably many labels. Taking $\phi(x) = x$ recovers the subset-coverage ordering. Taking $\phi(x) = 110x/(x+5)$ and $k = 50$ gives the corresponding fixed-category analogue of the MotifBench subset score. Appendix F.9 gives the residual bridge from this fixed-category utility to the official set-dependent Foldseek readout.

E.10 Tail evolution and remaining category signal

After the controlled window, let H be the complete state and Z the stratum label. The common continuation and evaluation kernel T_{tail} depends only on H, so $Z \to H \to O$ is a Markov chain. For conditional state laws μ_z and $\mu_{z'}$, define $\mathrm{TV}(\mu,\nu) = \sup_E |\mu(E) - \nu(E)|$. Then

$$\mathrm{TV}(\mu_z T_{\mathrm{tail}}, \mu_{z'} T_{\mathrm{tail}}) \leq \mathrm{TV}(\mu_z, \mu_{z'}). \tag{30}$$

Since $0 \leq T_{\mathrm{tail}}\mathbf{1}_E \leq 1$,

$$|\mu_z T_{\mathrm{tail}}(E) - \mu_{z'} T_{\mathrm{tail}}(E)| = \left|\int T_{\mathrm{tail}}\mathbf{1}_E \, d(\mu_z - \mu_{z'})\right| \leq \mathrm{TV}(\mu_z, \mu_{z'}).$$

Taking the supremum over E proves equation 30.

An identity kernel preserves all stratum information, whereas a state-independent kernel removes it completely. Thus the coverage advantage depends on layout or process strata retaining information about final successful categories after the common tail. This argument applies after the controlled window; during the window, unconsumed innovations depend on the process target. Pool-wide rollout reweighting depends on the complete parent pool and scores and is analyzed in Appendix F.

F Joint analysis of S^3-Scaffold

Appendix E studies the matched-reference parent pool before rollout reweighting. This appendix analyzes S^3-Scaffold after systematic allocation, when guidance has changed the parent states, EMPS has changed their joint distribution, and rollout reweighting has changed the number of offspring assigned to each parent. The exact identities below hold conditional on a realized parent pool and allocation. The coverage bounds incorporate explicit conditions on parent retention and successful continuation.

F.1 Complete parent states and continuation probabilities

Let $\mathcal{H} = (H_\tau^{(1)}, \ldots, H_\tau^{(M)})$ be the complete parent pool saved immediately before the $t = 10$ step, and let $\boldsymbol{a} = (a_1, \ldots, a_M)$ be its fresh four-sequence scores. With $\tau = 10$, each $H_\tau^{(i)}$ includes the sequence, self-conditioning state, layout, motif mapping, guidance records, and every persistent variable used by the continuation. Scoring leaves these snapshots unchanged; each offspring restores its parent's snapshot before installing a fresh continuation stream. Let $T_{\mathrm{tail}}(H_\tau^{(i)}, dy)$ be the fixed continuation kernel, including gated gradient guidance and final evaluation with eight fresh sequences. Conditional on the parent pool, scores, and systematic allocation, all offspring use independent fresh random tapes with this kernel. The kernel accesses the parent through $H_\tau^{(i)}$ and does not separately access its EMPS cell or other parents.

Fix a measurable, outputwise class map κ before sampling. Its values are $0, 1, 2, \ldots$, with 0 denoting failure. Write

$$q_{ic} = T_{\mathrm{tail}}\mathbf{1}_{\{\kappa=c\}}(H_\tau^{(i)}), \qquad q_{i,+} = \sum_{c>0} q_{ic}, \qquad \bar{a} = \frac{1}{M}\sum_i a_i > 0.$$

Writing $q_c(h) = T_{\mathrm{tail}}\mathbf{1}_{\{\kappa=c\}}(h)$ gives $q_{ic} = q_c(H_\tau^{(i)})$. For the matched one-continuation reference, $p_{ab,c} = \mathbb{E}[q_c(H_{\mathrm{par}}) \mid \mathcal{C}_{ab}]$, where H_{par} is one random parent state under that reference. Allocation uses equation 7, with $\lambda = 0.25$, and returns counts $\boldsymbol{n} = (n_i)_{i=1}^M$, where n_i is the number of offspring of parent i and $\sum_i n_i = M$. Let N_c be the number of offspring in class c, $S_+ = \sum_{c>0} N_c$, and $K_{\mathrm{full}} = \sum_{c>0} \mathbf{1}_{\{N_c>0\}}$.

F.2 Conditional expected success and coverage

Proposition (conditional coverage and copy increments). For every realized $\mathcal{H}, \boldsymbol{a}, \boldsymbol{n}$,

$$\mathbb{E}[K_{\mathrm{full}} \mid \mathcal{H}, \boldsymbol{a}, \boldsymbol{n}] = \sum_{c>0}\left\{1 - \prod_{i=1}^M (1-q_{ic})^{n_i}\right\}, \tag{31}$$

$$\mathbb{E}[S_+ \mid \mathcal{H}, \boldsymbol{a}, \boldsymbol{n}] = \sum_i n_i q_{i,+}. \tag{32}$$

Adding one fresh offspring of parent i changes the conditional expected coverage by

$$\Delta_i K(\boldsymbol{n}) = \sum_{c>0} q_{ic} \prod_{j=1}^M (1-q_{jc})^{n_j}, \tag{33}$$

and changes the conditional expected success count by $q_{i,+}$. Equation 33 increases the total budget by one. A transfer at fixed budget also removes an offspring from another parent.

Proof. Conditional independence gives

$$\Pr(N_c = 0 \mid \mathcal{H}, \boldsymbol{a}, \boldsymbol{n}) = \prod_i (1-q_{ic})^{n_i}.$$

Summing complementary probabilities proves equation 31; linearity gives equation 32. For the ith unit vector $\mathbf{e}_i$,

$$\prod_j (1-q_{jc})^{n_j} - \prod_j (1-q_{jc})^{n_j + (\mathbf{e}_i)_j} = q_{ic}\prod_j (1-q_{jc})^{n_j},$$

which proves equation 33 after summing over c. We take empty products and 0^0 to be one. Tonelli's theorem applies to the nonnegative sums, with $K_{\mathrm{full}} \le M$. □

F.3 Shadow one-continuation reference

For each realized parent pool, define a shadow procedure that gives every parent exactly one fresh continuation under the same T_{tail}, fixed class map, and final evaluation protocol. Let K_{sh} denote its successful-class coverage. This analytical reference is the post-parent analogue of the matched one-continuation reference in Appendix E.1 and isolates the effect of allocation.

F.4 Omitted-parent losses and extra-copy gains

Write

$$\mathcal{R} = \{i : n_i \ge 1\}, \qquad \mathcal{O} = \{i : n_i = 0\}, \qquad e_i = (n_i - 1)_+.$$

For each class, define the conditional deletion loss and extra-copy gain

$$L_c(\mathcal{H}, \boldsymbol{n}) = \prod_{i\in\mathcal{R}}(1-q_{ic})\left\{1 - \prod_{i\in\mathcal{O}}(1-q_{ic})\right\},$$

$$G_c(\mathcal{H}, \boldsymbol{n}) = \prod_{i\in\mathcal{R}}(1-q_{ic})\left\{1 - \prod_i (1-q_{ic})^{e_i}\right\}.$$

Both terms are nonnegative.

Proposition (exact shadow decomposition). For the shadow coverage defined above,

$$\mathbb{E}[K_{\mathrm{full}} - K_{\mathrm{sh}} \mid \mathcal{H}, \boldsymbol{a}, \boldsymbol{n}] = \sum_{c>0} (G_c - L_c).$$

For a design $R \in \{\mathrm{J}, \mathrm{L}, \mathrm{I}\}$, let

$$L_R = \mathbb{E}_R \sum_{c>0} L_c, \qquad G_R = \mathbb{E}_R \sum_{c>0} G_c.$$

These expectations include the parent pool, its scoring randomness, and the systematic offset. In particular, with $\Delta_{\mathrm{full}} = \mathbb{E}_{\mathrm{J}} K_{\mathrm{full}} - \mathbb{E}_{\mathrm{L}} K_{\mathrm{full}}$ and $\Delta_{\mathrm{sh}} = \mathbb{E}_{\mathrm{J}} K_{\mathrm{sh}} - \mathbb{E}_{\mathrm{L}} K_{\mathrm{sh}}$,

$$\Delta_{\mathrm{full}} = \Delta_{\mathrm{sh}} - (L_{\mathrm{J}} - L_{\mathrm{L}}) + (G_{\mathrm{J}} - G_{\mathrm{L}}).$$

Proof. For a fixed c, abbreviate the three products over retained parents, omitted parents, and extra copies by $P_{\mathcal{R}}$, $P_{\mathcal{O}}$, and P_e. The actual and shadow miss probabilities are $P_{\mathcal{R}} P_e$ and $P_{\mathcal{R}} P_{\mathcal{O}}$, so their coverage difference is

$$P_{\mathcal{R}}(P_{\mathcal{O}} - P_e) = P_{\mathcal{R}}(1 - P_e) - P_{\mathcal{R}}(1 - P_{\mathcal{O}}) = G_c - L_c.$$

Summation and expectation prove both identities, with integrability ensured by

$$\sum_{c>0} L_c \le \sum_{i \in \mathcal{O}} q_{i,+} \le M, \qquad \sum_{c>0} G_c \le \sum_i e_i q_{i,+} \le M.$$

□

The S^3-Scaffold difference depends on both the shadow gain and the differences in deletion and extra-copy terms.

F.5 Anchor retention and the allocation safety event

For the remaining results, let U denote one complete parent input, including its own fresh scoring randomness. Its reference law is μ, with $H = H(U)$ and score $a = a(U)$. The score is the unnormalized, parentwise quantity in equation 6; in the real-valued model, $0 < a \le 1$. Set $\mu_a = \mathbb{E}_\mu a$. The reference generator, including its complete random tapes, is assumed to be defined almost surely. These laws are taken before any conditioning on successful completion or final success.

The designs $\mathrm{J}, \mathrm{L}, \mathrm{I}$ share this reference law, the same continuation kernel, and the same evaluation protocol. For J, the $M = 100$ inputs are independent conditional draws from the 10×10 equal-mass layout–process cells. For L, there are ten independent draws from each of the ten layout strata. For I, all M inputs are independent draws from μ. Index inputs by $s = 1, \ldots, M$ before any random slot permutation. Let μ_s^R denote the input law at index s under design R. In each design, $M^{-1} \sum_s \mu_s^R = \mu$. Independence is used in this index order; the later slot permutation leaves the pool mean and a uniform subset's law unchanged.

Fix a finite class set $\mathcal{C}_0$, a common anchor-score threshold γ, constants ρ_c, and local events E_c such that

$$0 < \mu_a < \gamma \le 1, \qquad 0 < \rho_c \le 1, \qquad U \in E_c \implies a(U) \ge \gamma \quad \text{and} \quad q_c(H(U)) \ge \rho_c.$$

Each event is fixed in advance and depends only on that parent's complete state and own score. It does not depend on the pool mean, other parents, the allocation, or offspring outcomes. We call such a parent an anchor for class c. The events may overlap. The algorithm allocates continuations from parent scores; the anchor events specify analytical conditions on the frozen generator.

Define the safety event and its risk bound by

$$\mathcal{E}_{\mathrm{safe}} = \{\bar{a} \le \gamma\}, \qquad \sigma_{a,R}^2 = \mathrm{Var}_R(\bar{a}), \qquad d = \gamma - \mu_a > 0,$$
$$\bar{\epsilon}_R = \min\left\{ \exp(-2Md^2), \frac{\sigma_{a,R}^2}{\sigma_{a,R}^2 + d^2} \right\}.$$

Lemma (retention and safety risk). On $\mathcal{E}_{\text{safe}}$, every parent in $\bigcup_{c\in\mathcal{C}_0} E_c$ receives at least one offspring, for every systematic offset. Moreover,

$$\mathbb{P}_R(\mathcal{E}_{\text{safe}}^c) \le \bar{\epsilon}_R, \tag{34}$$
$$\sigma_{a,R}^2 = \frac{\mathrm{Var}_\mu(a)}{M} - \frac{1}{M^2}\sum_{s=1}^{M}\left(\mathbb{E}_{\mu_s^R} a - \mu_a\right)^2.$$

In particular, $\sigma_{a,\mathrm{J}}^2 \le \sigma_{a,\mathrm{L}}^2 \le \sigma_{a,\mathrm{I}}^2$ and $\bar{\epsilon}_{\mathrm{J}} \le \bar{\epsilon}_{\mathrm{L}} \le \bar{\epsilon}_{\mathrm{I}}$.

Proof. For an anchor on $\mathcal{E}_{\text{safe}}$, $a_i \ge \gamma \ge \bar{a}$, so

$$M\pi_i = \lambda + (1-\lambda)\frac{a_i}{\bar{a}} \ge 1. \tag{35}$$

The scaled systematic grid has spacing one and assigns parent i an interval of length $M\pi_i$, giving

$$n_i \in \{\lfloor M\pi_i\rfloor, \lceil M\pi_i\rceil\} \quad\Longrightarrow\quad n_i \ge 1.$$

For $a_s \in [0,1]$, the variance under exponential tilting is at most $1/4$. Integrating this bound twice gives $\log \mathbb{E}_R e^{t(a_s - \mathbb{E}_R a_s)} \le t^2/8$. Independence in cell order and the exponential Markov inequality imply

$$\mathbb{P}_R(\bar{a} - \mu_a > d) \le \inf_{t>0} e^{-tMd}\prod_s \mathbb{E}_R e^{t(a_s-\mathbb{E}_R a_s)} \le \inf_{t>0} e^{-tMd + Mt^2/8} = e^{-2Md^2}.$$

Alternatively, for $X = \bar{a} - \mu_a$, Markov's inequality gives

$$\mathbb{P}_R(X > d) \le \inf_{u\ge 0}\frac{\mathbb{E}_R(X+u)^2}{(d+u)^2} = \inf_{u\ge 0}\frac{\sigma_{a,R}^2 + u^2}{(d+u)^2} = \frac{\sigma_{a,R}^2}{\sigma_{a,R}^2 + d^2},$$

attained at $u = \sigma_{a,R}^2/d$, including when $\sigma_{a,R}^2 = 0$.

Independence and $M^{-1}\sum_s \mu_s^R = \mu$ give

$$\sigma_{a,R}^2 = \frac{1}{M^2}\sum_s \mathrm{Var}_{\mu_s^R}(a) = \frac{\mathrm{Var}_\mu(a)}{M} - \frac{1}{M^2}\sum_s (\mathbb{E}_{\mu_s^R} a - \mu_a)^2.$$

For $b_{uv} = \mathbb{E}[a \mid \mathcal{C}_{uv}]$ and $b_u = 10^{-1}\sum_v b_{uv}$,

$$\sum_{u,v}(b_{uv}-\mu_a)^2 - 10\sum_u (b_u - \mu_a)^2 = \sum_{u,v}(b_{uv} - b_u)^2 \ge 0.$$

For I, every conditional mean equals μ_a, proving the variance order. The risk order follows because $x/(x+d^2)$ is increasing and the exponential bound is common to all designs. □

F.6 Successful-continuation conditions and categorical coupling

The anchor condition supplies a class-specific lower bound on the probability of a successful continuation from a retained parent. We encode these lower bounds as categorical virtual outputs and couple them to the actual offspring.

Define, for every complete parent input,

$$r_c(U) = \rho_c \mathbf{1}_{E_c}(U), \qquad c \in \mathcal{C}_0.$$

The anchor condition implies $0 \le r_c(U) \le q_c(H(U))$. Since the actual classes are mutually exclusive, $\sum_{c\in\mathcal{C}_0} r_c(U) \le 1$, even when the events E_c overlap. Thus each parent admits one categorical virtual output Z_s, with conditional probability $r_c(U_s)$ for class c and the remaining probability for a null virtual label 0. This null label includes successful outputs removed by thinning or outside $\mathcal{C}_0$; it is distinct from an actual evaluation failure. Using one categorical output per parent preserves this exclusivity.

Lemma (full-slot coupling). There is a coupling of the virtual outputs, the M actual offspring, and uniform k-slot subsets, $1 \le k \le M$, each independent of its corresponding outputs. The virtual outputs are independent in cell index order, with probabilities

$$r^R_{s,c} = \rho_c h^R_{s,c}, \qquad h^R_{s,c} = \mathbb{P}_{\mu^R_s}(E_c);$$

and on $\mathcal{E}_{\mathrm{safe}}$, every class covered by the virtual subset is covered by the actual subset.

Proof. Give each of the M parents one virtual slot. Before drawing any tail randomness, choose a bijection ψ from virtual to actual offspring slots, conditional on the pool, permutation, and allocation. On $\mathcal{E}_{\mathrm{safe}}$, map each anchor to its first offspring, which exists by equation 35, and complete the map over the remaining slots. On $\mathcal{E}^c_{\mathrm{safe}}$, use any fixed bijection.

Let $Y_{\psi(s)}$ be the actual class in slot $\psi(s)$ and draw independent $V_s \sim \mathrm{Unif}[0,1]$. On $\mathcal{E}_{\mathrm{safe}}$, set

$$Z_s = \begin{cases} c, & Y_{\psi(s)} = c \in \mathcal{C}_0, \quad V_s \le r_c(U_s)/q_c(H(U_s)), \\ 0, & \text{otherwise}, \end{cases}$$

with a zero ratio when the denominator is zero. Nonanchors have $r_c(U_s) = 0$ for all c and hence $Z_s = 0$ almost surely. On $\mathcal{E}^c_{\mathrm{safe}}$, draw Z_s independently with probabilities $r_c(U_s)$ and $r_0(U_s) = 1 - \sum_{c\in\mathcal{C}_0} r_c(U_s)$. The distinct tail tapes and auxiliary uniforms give, in both cases,

$$\Pr(Z_1 = z_1, \dots, Z_M = z_M \mid (U_s)_{s=1}^M, \Pi, \boldsymbol{n}) = \prod_{s=1}^M r_{z_s}(U_s),$$

where Π is the slot permutation. Integrating over the independent inputs gives the stated marginal probabilities and independence.

Draw a uniform k-subset A of all M virtual slots independently of all other randomness. For every actual k-subset B, the bijection gives

$$\Pr(\psi(A) = B \mid (U_s)_{s=1}^M, \Pi, \boldsymbol{n}, (Y_j)_j, (Z_s)_s) = \binom{M}{k}^{-1}.$$

Thus $\psi(A)$ is uniform and independent of the actual outputs. On $\mathcal{E}_{\mathrm{safe}}$, $Z_s > 0$ implies $Z_s = Y_{\psi(s)}$, proving the coverage inclusion. □

F.7 Finite-batch lower bounds on expected coverage

Theorem (coverage and utility lower bounds). Let $K_{\mathrm{full},R,k}$ and $K_{\mathrm{v},R,k}$ count the actual and virtual classes in their uniform k-slot subsets. For a nondecreasing function ϕ on the nonnegative integers with $\phi(0) = 0$, define

$$H_{R,k} = \mathbb{E}_R \phi(K_{\mathrm{v},R,k}).$$

Then

$$\mathbb{E}_R \phi(K_{\mathrm{full},R,k}) \ge \left[H_{R,k} - \phi(k)\bar{\epsilon}_R\right]_+. \tag{36}$$

The factor $\phi(k)$ can be replaced by $\phi(\min\{k, |\mathcal{C}_0|\})$. For one fixed class c, the corresponding hit bound is

$$\mathbb{P}_R(N_{k,c} > 0) \ge \left[1 - \frac{e_k(1-\rho_c h^R_{1,c}, \dots, 1-\rho_c h^R_{M,c})}{\binom{M}{k}} - \bar{\epsilon}_R\right]_+, \tag{37}$$

where $e_k(x_1, \dots, x_M) = \sum_{|A|=k} \prod_{s\in A} x_s$. For the full batch, this becomes

$$\mathbb{P}_R(N_c > 0) \ge \left[1 - \prod_{s=1}^M (1 - \rho_c h^R_{s,c}) - \bar{\epsilon}_R\right]_+. \tag{38}$$

Proof. The coupling and $0 \le K_{\mathrm{v},R,k} \le \min\{k, |\mathcal{C}_0|\}$ give

$$\phi(K_{\mathrm{full},R,k}) \ge \phi(K_{\mathrm{v},R,k}) - \phi(\min\{k, |\mathcal{C}_0|\})\mathbf{1}_{\mathcal{E}^c_{\mathrm{safe}}}.$$

Taking expectations, applying equation 34, and using nonnegativity proves equation 36 and its sharper factor.

For a single class, let $N^{\mathrm{v}}_{k,c}$ count its virtual occurrences in the subset. Independence gives

$$\Pr(N^{\mathrm{v}}_{k,c} = 0) = \frac{1}{\binom{M}{k}} \sum_{|A|=k} \prod_{s \in A} (1 - \rho_c h^R_{s,c}) = \frac{e_k(1 - \rho_c h^R_{1,c}, \ldots, 1 - \rho_c h^R_{M,c})}{\binom{M}{k}}.$$

The coupling also gives

$$\mathbf{1}_{\{N_{k,c}>0\}} \ge \mathbf{1}_{\{N^{\mathrm{v}}_{k,c}>0\}} - \mathbf{1}_{\mathcal{E}^c_{\mathrm{safe}}}.$$

Taking expectations proves equation 37. Taking $k = M$ gives equation 38. □

F.8 How stratification improves the common lower bound

The same anchors, threshold, class map, continuation kernel, and subset protocol are used for all three designs. If ϕ is also concave on the integers, then

$$H_{\mathrm{J},k} \ge H_{\mathrm{L},k} \ge H_{\mathrm{I},k}. \tag{39}$$

Combining this with the safety-risk order gives

$$[H_{\mathrm{J},k} - \phi(k)\bar{\epsilon}_{\mathrm{J}}]_+ \ge [H_{\mathrm{L},k} - \phi(k)\bar{\epsilon}_{\mathrm{L}}]_+ \ge [H_{\mathrm{I},k} - \phi(k)\bar{\epsilon}_{\mathrm{I}}]_+ . \tag{40}$$

Proof. The virtual laws satisfy

$$r^{\mathrm{L}}_{u,c} = \frac{1}{10} \sum_v r^{\mathrm{J}}_{uv,c}, \qquad r^{\mathrm{I}}_c = \frac{1}{10} \sum_u r^{\mathrm{L}}_{u,c}.$$

Applying equation 29 within layout rows, then across all slots, proves equation 39. The reverse risk order and monotonicity of $[\cdot]_+$ give equation 40. □

Equation 40 compares the common lower bounds for the three sampling designs.

F.9 Bridge to the official set-dependent clustering score

The official readout clusters the successful outputs of the full batch, then samples k of its M slots and inherits the full-batch labels. Let $F_{R,k}$ be the number of these labels represented in the subset. Here $F_{R,k}$ uses labels determined from the full successful set, while $K_{\mathrm{full},R,k}$ uses the fixed outputwise map. On the same batch and subset, define

$$\delta_{R,k} = \mathbb{E}_R \left| F_{R,k} - K_{\mathrm{full},R,k} \right| .$$

This expectation includes both batch generation and subset selection. We do not estimate or control $\delta_{R,k}$ in the reported experiments. The inequalities below express how control of this residual transfers fixed-category utilities to the official readout. For the score $\phi(x) = 110x/(x+5)$ and every nonnegative integer j,

$$\phi(j+1) - \phi(j) = \frac{550}{(j+5)(j+6)} \le \frac{55}{3}.$$

Summing these increments and taking expectations gives

$$\begin{aligned} |\mathbb{E}_R \phi(F_{R,k}) - \mathbb{E}_R \phi(K_{\mathrm{full},R,k})| &\le \mathbb{E}_R \left| \phi(F_{R,k}) - \phi(K_{\mathrm{full},R,k}) \right| \\ &\le \frac{55}{3} \delta_{R,k}. \end{aligned}$$

In particular, $\phi(50) = 100$, so equation 36 implies

$$\mathbb{E}_R \phi(F_{R,50}) \ge \left[H_{R,50} - 100\bar{\epsilon}_R - \frac{55}{3}\delta_{R,50} \right]_+ .$$

For a direct comparison, the same absolute-error bounds give

$$\begin{aligned} \mathbb{E}_{\mathrm{J}} \phi(F_{\mathrm{J},k}) - \mathbb{E}_{\mathrm{L}} \phi(F_{\mathrm{L},k}) \ge \mathbb{E}_{\mathrm{J}} \phi(K_{\mathrm{full,J},k}) - \mathbb{E}_{\mathrm{L}} \phi(K_{\mathrm{full,L},k}) \\ - \frac{55}{3}(\delta_{\mathrm{J},k} + \delta_{\mathrm{L},k}). \end{aligned}$$

This comparison uses the actual fixed-class score difference together with bounds on the two clustering residuals.

F.10 INTERPRETATION OF THE JOINT ANALYSIS

Gated gradient guidance changes the realized parent states and hence the continuation probabilities q_{ic}; rollout reweighting changes the offspring counts n_i; and EMPS changes the joint law of the parent pool. Equations 31–32 and the shadow decomposition are exact, whereas equations 36–40 are lower bounds under the stated anchor, reachability, and safety conditions. The latter compare Joint, Layout-only, and IID under common conditions. The pre-reweighting comparison in Appendix E also includes Process-only.

The anchor probabilities, reachability constants ρ_c, and threshold gap $\gamma - \mu_a$ remain unestimated in the reported experiments and serve as analytical sufficient-condition parameters. For the official set-dependent Foldseek score, Appendix F.9 relates each fixed-category utility to its corresponding readout through the residual $\delta_{R,k}$. Within this bridge, comparisons of actual complete-system scores use the fixed-category utility difference together with bounds on the clustering residuals.

G COMPLETE EXPERIMENTAL EVIDENCE

This appendix consolidates the endpoint tables, gating comparison, coverage calculations, selected occupancy illustration, plotting protocols, and public-baseline provenance supporting Section 4. Unless stated otherwise, each in-house configuration consists of one saved pool of 100 designs for each of the 30 tasks. The exact calculations average uniform subset selection conditional on each saved pool.

G.1 FULL-BENCHMARK SIX-CONFIGURATION RESULTS

Tables 13–15 report every task for the six configurations in Table 2. Each task pool contains 100 designs, evaluated with eight sequences per design. The columns use the same configuration names and order as Table 2; the leftmost method column is frozen RFdiffusion, and the rightmost is S^3-Scaffold. Zero-success tasks remain in every table.

All per-task scores here are exact expectations of the official 50-of-100 statistic for the saved pools and their original cluster assignments, as defined in Appendix A.5. The RFdiffusion column is an exact re-reading of the public archive, with mean 21.9182; the public headline value 21.95 in Tables 1 and 2 is unchanged. Successful-cluster counts refer to the full 100-design pools. For S^3-Scaffold w/o EMPS, we retain the same saved clustering used by the main component table, totaling 214 clusters.

Table 13: Per-task full-benchmark results (a of c): exact 50-of-100 scores for the six configurations in Table 2. Shading marks S^3-Scaffold and its direct base. The RFdiffusion mean re-read from the public archive is 21.9182; its public headline Score remains 21.95.

Task	RFdiffusion (base)	Gated gradient guidance	Rollout reweighting	EMPS	S^3-Scaffold w/o EMPS	S^3-Scaffold (ours)
01_1LDB	17.0370	9.1667	28.9881	17.0370	17.0370	17.0370
02_1ITU	24.4426	17.2566	42.3347	26.4531	28.9881	22.9860
03_2CGA	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
04_5WN9	23.8492	35.0072	59.1977	23.8492	57.5028	35.5414
05_5ZE9	85.6698	86.1316	81.4517	82.6610	84.4803	88.5627
06_6E6R	93.3402	91.8679	93.9539	94.1711	91.8112	95.1336
07_6E6R	95.4456	97.6141	96.3114	95.9857	97.7868	97.3961
08_7AD5	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
09_7CG5	87.1283	85.9731	87.0908	88.6666	89.2722	84.2084
10_7WRK	9.1667	0.0000	16.1111	13.7963	17.0370	17.0370
11_3TQB	36.3390	46.9576	48.4556	42.9121	31.3469	54.4282
12_4JHW	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
13_4JHW	0.0000	9.1667	0.0000	0.0000	17.0370	17.0370
14_5IUS	28.1871	24.8718	18.3333	34.9564	31.3468	31.4271
15_7A8S	30.4818	17.8174	31.3908	34.9412	31.3466	36.3037
16_7BNY	21.0053	34.9959	38.6856	17.0370	24.4426	37.5950
17_7DGW	18.3333	28.1878	18.3333	18.3333	18.3333	18.3333
18_7MQQ	9.1667	13.7963	17.2566	13.7963	21.0053	17.2566
19_7MQQ	29.7900	9.1667	34.9496	45.0890	44.2900	38.0843
20_7UWL	0.0000	0.0000	0.0000	0.0000	9.1667	9.1667
21_1B73	0.0000	0.0000	0.0000	0.0000	22.9860	22.9860
22_1BCF	18.3333	18.3333	18.3333	18.3333	18.3333	18.3333
23_1MPY	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
24_1QY3	0.0000	23.8492	9.1667	0.0000	38.1078	52.5785
25_2RKX	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
26_3B5V	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
27_4XOJ	0.0000	0.0000	0.0000	0.0000	27.3489	42.3347
28_5YUI	29.8311	47.1335	48.7135	36.9173	46.0220	54.4301
29_6CPA	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
30_7UWL	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Mean score	21.9182	23.2431	26.3019	23.4979	28.8343	30.2732

Table 14: Per-task full-benchmark results (b of c): successful designs in each 100-design task pool for the six configurations in Table 2. Shading marks S^3-Scaffold and its direct base. All 30 tasks are retained; the final row counts solved tasks.

Task	RFdiffusion (base)	Gated gradient guidance	Rollout reweighting	EMPS	S^3-Scaffold w/o EMPS	S^3-Scaffold (ours)
01_1LDB	2	1	5	2	2	2
02_1ITU	6	4	8	5	5	4
03_2CGA	0	0	0	0	0	0
04_5WN9	3	5	15	3	17	6
05_5ZE9	41	41	52	36	57	63
06_6E6R	75	74	89	70	89	82
07_6E6R	68	79	84	71	90	90
08_7AD5	0	0	0	0	0	0
09_7CG5	70	70	78	63	87	77
10_7WRK	1	0	3	2	2	2
11_3TQB	46	48	70	50	81	82
12_4JHW	0	0	0	0	0	0
13_4JHW	0	1	0	0	2	2
14_5IUS	15	11	22	19	24	31
15_7A8S	10	5	26	13	21	20
16_7BNY	3	7	12	2	6	8
17_7DGW	30	40	28	25	47	47
18_7MQQ	1	2	4	2	3	4
19_7MQQ	4	1	14	11	11	10
20_7UWL	0	0	0	0	1	1
21_1B73	0	0	0	0	4	4
22_1BCF	100	100	100	100	100	100
23_1MPY	0	0	0	0	0	0
24_1QY3	0	3	1	0	7	11
25_2RKX	0	0	0	0	0	0
26_3B5V	0	0	0	0	0	0
27_4XOJ	0	0	0	0	6	8
28_5YUI	6	14	15	7	22	24
29_6CPA	0	0	0	0	0	0
30_7UWL	0	0	0	0	0	0
Total	481	506	626	481	684	678
Solved /30	17	18	18	17	22	22

Table 15: Per-task full-benchmark results (c of c): successful clusters in each full 100-design task pool for the six configurations in Table 2. Shading marks S^3-Scaffold and its direct base; each configuration retains its original saved clustering.

Task	RFdiffusion (base)	Gated gradient guidance	Rollout reweighting	EMPS	S^3-Scaffold w/o EMPS	S^3-Scaffold (ours)
01_1LDB	2	1	3	2	2	2
02_1ITU	2	1	5	2	3	2
03_2CGA	0	0	0	0	0	0
04_5WN9	3	5	10	3	8	4
05_5ZE9	32	33	22	28	23	29
06_6E6R	50	44	47	53	39	51
07_6E6R	65	79	60	67	73	67
08_7AD5	0	0	0	0	0	0
09_7CG5	31	27	26	32	32	23
10_7WRK	1	0	1	1	2	2
11_3TQB	3	4	5	4	2	5
12_4JHW	0	0	0	0	0	0
13_4JHW	0	1	0	0	2	2
14_5IUS	2	2	1	3	2	2
15_7A8S	2	1	2	3	2	3
16_7BNY	2	4	3	2	2	4
17_7DGW	1	2	1	1	1	1
18_7MQQ	1	1	1	1	2	1
19_7MQQ	4	1	3	5	5	4
20_7UWL	0	0	0	0	1	1
21_1B73	0	0	0	0	2	2
22_1BCF	1	1	1	1	1	1
23_1MPY	0	0	0	0	0	0
24_1QY3	0	3	1	0	4	8
25_2RKX	0	0	0	0	0	0
26_3B5V	0	0	0	0	0	0
27_4XOJ	0	0	0	0	2	5
28_5YUI	3	6	5	4	4	6
29_6CPA	0	0	0	0	0	0
30_7UWL	0	0	0	0	0	0
Total	205	216	197	212	214	225

G.2 Full-benchmark gating comparison

Table 16 compares ungated and gated guidance on the RFdiffusion background and after adding rollout reweighting. These are four separately generated full-benchmark pools. Without rollout reweighting, gating changes Score, S_+, and successful clusters from $22.4622/483/215$ to $23.2431/506/216$, while solved tasks change from 19 to 18. With rollout reweighting, all four metrics improve from $26.4548/20/597/202$ to $28.8343/22/684/214$.

The two ungated Scores are archived estimates based on 1,000 uniform 50-of-100 subsets per task. The underlying ungated cluster occupancies are unavailable. The two gated Scores are exact expectations under the same subset statistic. Table 16 identifies the scoring convention for each configuration.

Table 16: Full-benchmark comparison of ungated and gated geometric guidance across separate saved pools. Bold marks the better value within each no-rollout or with-rollout pair, except that ungated guidance alone solves one additional task.

Configuration	Score ↑	Solved /30 ↑	S_+ /3,000 ↑	Successful clusters ↑
Ungated gradient guidance	22.4622^{B}	**19**	483	215
Gated gradient guidance	$\mathbf{23.2431}^{\mathrm{E}}$	18	**506**	**216**
Ungated guidance + rollout reweighting	26.4548^{B}	20	597	202
S^3-Scaffold w/o EMPS	$\mathbf{28.8343}^{\mathrm{E}}$	**22**	**684**	**214**

Each row is one 3,000-design pool. ^{B}Archived estimate from 1,000 uniform 50-of-100 subsets per task. ^{E}Exact expectation of the same subset statistic conditional on the saved pool and clustering.

G.3 Coverage after composition

Table 17 gives the values underlying Figure 3(a,b). On RFdiffusion, EMPS leaves the observed successful-design count unchanged at 481 while increasing full-pool clusters from 205 to 212 and expected 50-design coverage from 116.5442 to 123.4417. After gated gradient guidance and rollout reweighting, EMPS increases full-pool clusters from 214 to 225 and expected 50-design coverage from 138.5412 to 147.6433, although the observed successful-design count changes from 684 to 678. Both comparisons use separately clustered saved pools, with expectations over uniform subsets.

Table 17: Successful-structure coverage before and after adding EMPS to RFdiffusion and S^3-Scaffold w/o EMPS. All quantities are computed from one saved 100-design pool per task.

Configuration	S_+ /3,000	$\mathbb{E}[K_{50}]$	K_{100}
RFdiffusion	481	116.5442	205
EMPS	481	**123.4417**	**212**
S^3-Scaffold w/o EMPS	**684**	138.5412	214
S^3-Scaffold (ours)	678	**147.6433**	**225**

K_{100} is the total number of successful clusters in the 30 complete pools. $\mathbb{E}[K_{50}]$ is the exact expected total number represented after independently drawing 50 of 100 designs without replacement from every task pool, conditional on the full-pool cluster assignments.

G.4 Single-axis occupancy illustration

Table 18 and Figure 4 examine the joint use of the two stratification axes on 14_5IUS, a selected illustrative task from the 30-task MotifBench benchmark. Layout-only stratifies Q, Process-only stratifies S, and Joint EMPS stratifies $Q \times S$. All three use the same registered native root and endpoint sequence-design seeds for each task and slot; rollout reweighting and gated gradient guidance are disabled. RFdiffusion uses its public result pool. Each configuration has 100 designs per task. Process-only produces one successful cluster, whereas Layout-only and Joint EMPS each produce three. Layout-only has occupancies $[18, 1, 1]$, while Joint EMPS has $[15, 3, 1]$; the larger second cluster raises exact expected 50-design coverage from 2.0000 to 2.3788 despite one fewer successful design. This selected task illustrates how more balanced occupancy lowers the probability of omitting a successful category under finite subsampling. In the occupancy illustration, cluster rank is defined separately within each pool. Figure 3(a) separately reports expected coverage across subset sizes within the saved pools over all 30 tasks.

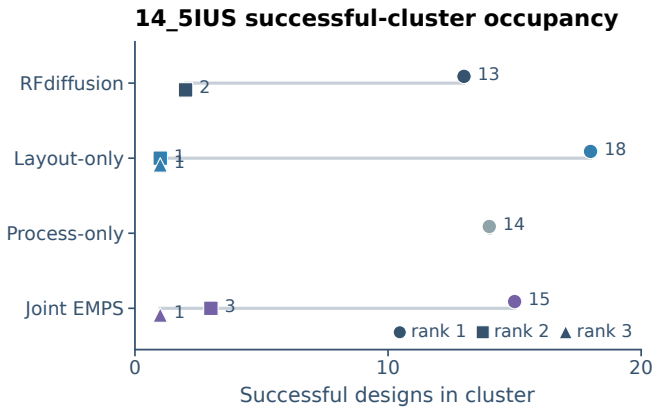

Figure 4: Successful-cluster occupancies for the selected 14_5IUS illustration. Clusters are ranked by decreasing occupancy separately within each pool.

Table 18: Single-task EMPS occupancy illustration on 14_5IUS, a selected task from the 30-task MotifBench benchmark. Each configuration uses frozen RFdiffusion and 100 designs.

Configuration	Score ↑	Successful designs S_+ ↑	Successful clusters K_{100} ↑	Expected clusters $\mathbb{E}[K_{50}]$ ↑
RFdiffusion	28.19	15	2	1.7525
Layout-only	30.62	**20**	**3**	2.0000
Process-only	18.33	14	1	1.0000
Joint EMPS	**34.96**	19	**3**	**2.3788**

Layout-only stratifies Q; Process-only stratifies S; Joint EMPS stratifies $Q \times S$. Rollout reweighting and gated gradient guidance are disabled. Score and $\mathbb{E}[K_{50}]$ are exact expectations over uniform 50-design subsets of each saved pool. The RFdiffusion row uses its public result pool.

Layout-only and Joint EMPS have successful-cluster occupancies $[18, 1, 1]$ and $[15, 3, 1]$, respectively. Bold values mark column maxima.

This selected example illustrates category omission under finite subsampling.

G.5 Diagnostic and plotting protocols

The plotting values for Figures 2, 3, and 4 are stored as lightweight CSV files under `data/figure2/` and `data/figure3/`. The coverage-curve CSV contains plotting values recovered from the archived vector figure, as documented in `data/README.md`. The corresponding deterministic plotting scripts are `scripts/plot_diagnostic_responses.py` and `scripts/plot_emps_coverage.py`. They set the plotting style, panel geometry, labels, colors, and PDF font embedding used in the manuscript. The scripts assert the headline counts and selected numerical values before writing the figures.

Generated geometry and endpoint success. Figure 2(a) includes all 3,000 public RFdiffusion designs, with 100 designs for each of the 30 MotifBench tasks. Generated-motif geometry is the NCACO RMSD to the original target after one joint Kabsch alignment across all motif fragments. A design succeeds if at least one of its eight evaluated sequences simultaneously satisfies scaffold-consistency error below 2 Å and refolded-motif RMSD below 1 Å, using the saved benchmark evaluation values. Success fractions are computed within the displayed geometry bins, with left-inclusive and right-exclusive boundaries and an unbounded final bin. All 3,000 designs are retained; no task or design is filtered after bin assignment.

Current-step guidance response. Figure 2(b) uses ungated translation–rotation guidance from earlier runs on eight trajectories, two per task, probed at $t \in \{50, 24, 14, 5, 1\}$. Of these 40 states, 32 have positive guidance budgets; the eight $t = 50$ probes are zero-budget controls. Each comparison restores the same noisy state, sequence, self-conditioning history, and random state. The shadow

evaluation omits only the current guidance update, while retaining the previously guided history. The plotted values are generated-motif NCAC RMSDs of the raw clean predictions. Guidance reduces this error in 31 of the 32 positive-budget states using a 10^{-5} Å directional threshold. These are repeated local probes along eight trajectories. All 40 probes are archived; the figure displays the 32 positive-budget probes and omits the eight zero-budget controls. The remaining positive-budget probe increases the error. This diagnostic measures finite-step responses at the recorded states of the earlier ungated trajectories.

Coverage within saved pools. Figures 3 and 4 condition on the saved 100-design pool and its full-pool successful-design clustering for each task and method. If successful cluster c contains n_{dc} designs in task d, uniform sampling of m designs without replacement gives the total expected coverage

$$\sum_{d=1}^{30}\sum_{c=1}^{K_d}\left[1-\frac{\binom{100-n_{dc}}{m}}{\binom{100}{m}}\right], \qquad m\in\{0,\dots,100\},$$

with $\binom{n}{m}=0$ when $m>n$. For Figure 3, the sum retains all 30 tasks, including those with no successful designs; the single-task illustration uses only the 14_5IUS term. The binomial ratio is the probability of omitting cluster c; its complement is the probability of representing that cluster at least once. Thus the observed cluster occupancies determine expected subset coverage. This identity explains the coverage readout for any saved pool without assuming how that pool was generated. The coverage curves and the $\mathbb{E}[K_{50}]$ column of Table 17 report exact conditional subsampling expectations using each saved pool's full-pool Foldseek labels. The 5IUS occupancy illustration orders successful clusters by decreasing size separately within each pool. No smoothing, fitted curve, or Monte Carlo approximation is used in either coverage panel.

G.6 Public-baseline provenance and score versions

Table 19 records the publication citation, permanent result archive, aggregation unit, and official Score used for every public baseline in Table 1. We retain the public leaderboard values at their reported precision. Consequently, the table reports the leaderboard values 31.14 for Genie3 and 29.75 for La-Proteina, although arithmetic means of the available two-decimal run summaries would conventionally round to 31.15 and 29.76. Similarly, RFdiffusion's official 21.95 is retained in the main comparison, while its archived cluster occupancies give the exact fixed-pool expectation 21.9182 reported in Table 13.

Table 19: Provenance of the public baselines in Table 1. The source column gives the permanent MotifBench result archive; publication details are supplied by the cited paper.

Method and publication	Leaderboard/archive source	Reported unit	Official Score
RFdiffusion3 (Butcher et al., 2025)	Zenodo 22112445	Three-run mean	43.52
Protpardelle-1c (cc91-large) (Lu et al., 2025)	Zenodo 19828416	One submitted pool	33.81
Genie3 (Lin et al., 2026)	Zenodo 21131140	Three-run mean	31.14
La-Proteina (Geffner et al., 2026)	Zenodo 21272894	Three-run mean	29.75
ODesign-rigid (Zhang et al., 2025)	Zenodo 19201706	One submitted pool	29.39
Protpardelle-1c (Lu et al., 2025)	Zenodo 15612060; aggregate 22042024	One submitted pool	27.83
Proteina (Geffner et al., 2025)	Zenodo 19204034	One submitted pool	25.46
RFdiffusion (Watson et al., 2023)	Zenodo 16718874	One submitted pool	21.95
Genie2 (Lin et al., 2024)	Zenodo 20394938	One submitted pool	20.83
FrameFlow (Yim et al., 2024)	Zenodo 21273108	Three-run mean	18.71

Values were read on 2026-09-26 from the public MotifBench leaderboard at revision `d5a714c`. The current scoring script reports the mean of 1,000 unseeded uniform 50-of-100 subset scores per task. Rows marked "three-run mean" average three independently generated submission pools; the remaining rows report one submitted pool. For Protpardelle-1c, the current leaderboard value is 27.83; 27.87 is the historical full-pool score retained in older archive metadata.