REFLEX: RNA Ensemble Generation via Flexibility-Calibrated Stochastic Bridge

Wanli Ma[1,2,3] **Tianmeng Hu**[1,2,3] **Ke Li**[1,2,3,4]*
[1]Earlham Institute, United Kingdom.
[2]Living Systems Institute, University of Exeter, United Kingdom.
[3]Department of Computer Science, University of Exeter, United Kingdom.
[4]School of Computing Sciences, University of East Anglia, United Kingdom.
Email: wm385@exeter.ac.uk, tianmeng0824@gmail.com,
Ke.Li@earlham.ac.uk

Abstract

RNA molecules often function through multiple experimentally observed conformational states. A central challenge is to capture ensemble diversity without sacrificing structural fidelity: samples may collapse toward averaged representative structures or drift into unsupported conformational regions. We introduce REFLEX (RNA Ensemble generation calibrated by FLEXibility), a sequence-conditioned generator for RNA backbone-frame ensembles based on a heteroscedastic stochastic bridge on $\mathrm{SE}(3)$. During training, experimental B-factors calibrate residue-wise bridge widths, exposing each observed conformer through local perturbation neighborhoods with residue-specific scales while keeping constrained regions sharper. A learned flexibility condition guides the bridge-field network to recover target conformer-specific frames from these calibrated noisy states. At inference, REFLEX uses only the input sequence, requiring no MSA-subsampling heuristics. On a multi-conformer RNA benchmark, REFLEX achieves the best precision–recall trade-off for ensemble coverage across evaluation thresholds, while recovery-stress ablations indicate gains in target-frame recovery under stronger perturbations.

1 Introduction

RNA molecules are intrinsically dynamic, often interconverting among multiple conformational states. Such ensembles are central to many RNA functions, including ligand recognition, molecular switching, and regulatory control [1–4]. Sequence-conditioned RNA modeling should therefore move beyond predicting one representative structure and aim to recover multiple plausible conformational states while preserving the local constraints that make each state structurally valid.

Recent deep learning methods have substantially advanced RNA 3D modeling. AlphaFold3 [5] and RhoFold+ [6] achieve strong accuracy when a single representative conformation is sufficient. Alternative conformations can sometimes be elicited by repeated inference under perturbed evolutionary context, such as MSA subsampling [7], but these procedures rely on inference-time search to explore conformational alternatives. Dedicated generative methods have begun to model RNA ensembles more directly: RNAbpFlow [8] generates RNA backbones conditioned on sequence and predicted base-pair information, while DynaRNA [9] models conformational ensembles through coordinate-space diffusion from structural inputs. Yet existing approaches do not explicitly calibrate how conformational uncertainty should vary across nucleotides.

*Corresponding author

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

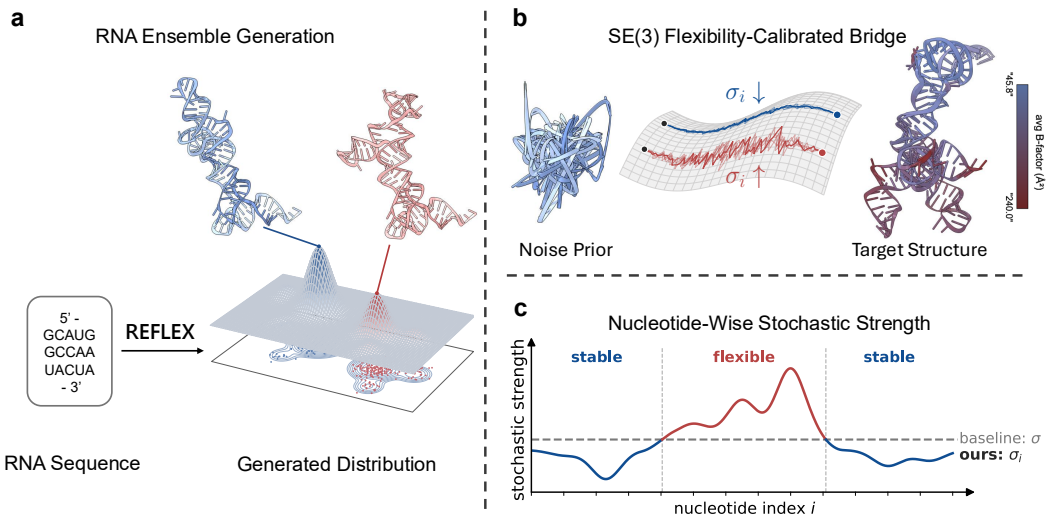


Figure 1: **Overview of** REFLEX**. a**: RNA ensemble generation aims to recover accurate and diverse conformational alternatives. **b**: Flexibility-calibrated stochastic bridge on $\mathrm{SE}(3)$. The structures on the right are multiple target conformations, and their averaged B-factor profile captures nucleotide-wise flexibility used to calibrate the stochastic bridge. **c**: Nucleotide-wise stochastic strength, with nucleotides on the horizontal axis and stochasticity strength on the vertical axis; flexible positions receive stronger stochasticity than constrained positions.

This local calibration matters because RNA flexibility is spatially heterogeneous. Base-paired helices are often constrained, whereas bulges, loops, junctions, and hinge regions can undergo larger-amplitude motion [10, 11, 3]. Thus, the same sequence can have both rigid regions with narrow coordinate support and dynamic regions with multiple plausible conformational targets. Without a training signal for this local ambiguity, stochastic generators face a precision–recall tension: they may collapse toward dominant conformations or spread samples into unsupported structures. A sequence-to-ensemble generator should therefore learn where structural uncertainty is expected and allocate it as a nucleotide-wise property of the molecule.

We introduce REFLEX, a sequence-conditioned stochastic bridge on $\mathrm{SE}(3)$ for RNA backbone-frame ensemble generation. During training, REFLEX uses chain-normalized experimental B-factors as a proxy for relative local mobility and converts them into nucleotide-wise bridge widths. Flexible nucleotides receive broader bridge perturbations, while rigid regions receive narrower ones. At inference, a sequence-derived latent flexibility condition guides the generative dynamics, incorporating learned relative-mobility information into generation. Figure 1 summarizes the sequence-to-ensemble task and the flexibility-calibrated bridge used by REFLEX.

Our main contributions are:

- We formulate RNA sequence-to-ensemble generation as stochastic bridge learning on $\mathrm{SE}(3)$, where the bridge field is trained to recover conformers from noisy frame states.
- We introduce a flexibility-calibrated heteroscedastic bridge that uses chain-normalized experimental B-factor to set nucleotide-wise bridge widths during training, while providing a sequence-derived latent flexibility condition to the bridge-field network at inference.
- We show on a multi-conformer RNA benchmark that REFLEX improves ensemble coverage, structural fidelity, and precision–recall balance over MSA-subsampling predictors and dedicated ensemble generators while maintaining meaningful structural diversity.

2 Preliminaries

2.1 RNA Representation and Problem Setup

Given an RNA sequence s of length L, let $\mathcal{S}_{\exp}(s) = \{G_m^\star\}_{m=1}^{M_s}$ denote the available experimental state representatives for this sequence. We do not assume access to calibrated thermodynamic populations. Our goal is to learn a conditional generator $p_\theta(G \mid s)$ whose samples $\{G^{(k)}\}_{k=1}^{K}$ collectively recover this observed conformational support while remaining individually plausible.

Following RNA-FRAMEFLOW [12], we represent the frame state of nucleotide i as a rigid transformation $G_i = (x_i, R_i) \in \mathrm{SE}(3)$, where $\mathrm{SE}(3) = \mathbb{R}^3 \rtimes \mathrm{SO}(3)$. Here $x_i \in \mathbb{R}^3$ is the position of the C4′ atom, and $R_i \in \mathrm{SO}(3)$ specifies the local orientation defined by the C3′–C4′–O4′ sugar geometry. We write $G = (G_1, \ldots, G_L) \in \mathrm{SE}(3)^L$ for the RNA backbone-frame state. We reconstruct all heavy-atom coordinates, including complete nucleobases, using ten torsion angles $\Phi_i \in (S^1)^{10}$, represented by sine–cosine pairs, and nucleotide-specific reference geometry (Appendix F). In REFLEX, the stochastic bridge is defined over the frame state G; torsions are predicted as auxiliary target-structure outputs for all-atom reconstruction.

2.2 $\mathrm{SE}(3)$ Flow Matching

Flow matching [13–15] learns a velocity field that transports a tractable prior p_0 to a data distribution by integrating an ordinary differential equation along a prescribed probability path. For a single nucleotide frame in $\mathrm{SE}(3)$, let (G_0, G_1) be a coupled pair of prior and data states, and let $t \in [0, 1]$. Conditional flow matching constructs a deterministic intermediate state by geodesic interpolation:

$$x_t = (1-t)x_0 + tx_1, \qquad R_t = R_0 \exp\big(t \log(R_0^\top R_1)\big)\,. \tag{1}$$

Differentiating this path yields closed-form conditional velocity targets

$$u_t^x = \frac{x_1 - x_t}{1-t}, \qquad u_t^R = \frac{\log(R_t^\top R_1)^\vee}{1-t} \in \mathbb{R}^3, \tag{2}$$

where $(\cdot)^\vee$ identifies an element of $\mathfrak{so}(3)$ with its Euclidean coordinate vector. A network predicting $(\hat{v}_t^x, \hat{v}_t^R)$ is trained by tangent-space regression,

$$\mathcal{L}_{\mathrm{SE}(3)}(\theta) = \mathbb{E}_{t, G_0, G_1}\left[\frac{1}{|\mathcal{R}|}\sum_{i \in \mathcal{R}}\left(\lambda_x \|\hat{v}_{t,i}^x - u_{t,i}^x\|_2^2 + \lambda_R \|\hat{v}_{t,i}^R - u_{t,i}^R\|_2^2\right)\right], \tag{3}$$

where $\mathcal{R}$ denotes resolved frame nucleotides and $\lambda_x, \lambda_R > 0$ balance translational and rotational terms. REFLEX uses this $\mathrm{SE}(3)$ velocity parameterization as a reference, but replaces the deterministic geodesic path with stochastic bridge marginals whose widths vary across nucleotides.

2.3 RNA Flexibility and B-factors

B-factors provide a widely available but noisy experimental proxy for local positional uncertainty [16]. In X-ray crystallography, they are refined as atomic displacement parameters and are related, under an ideal isotropic model, to mean-squared positional displacement [10]. In NMR-derived structures, analogous B-factor-like quantities can be estimated from coordinate dispersion across deposited structural ensembles, although these quantities are not identical to crystallographic B-factors [17, 18]. Because B-factors also depend on experimental modality, refinement, resolution, and static disorder, we do not treat their absolute values as calibrated thermodynamic mobilities across structures. Instead, they are best used as within-chain relative mobility signals, following prior computational studies of biomolecular flexibility [11, 19].

3 Proposed Method

REFLEX is a sequence-conditioned stochastic-bridge generator for RNA backbone-frame ensembles on $\mathrm{SE}(3)$. Building on $\mathrm{SE}(3)$ flow matching [20], REFLEX introduces a residue-wise heteroscedastic bridge whose local width is calibrated by relative B-factor mobility. This flexibility-calibrated bridge exposes mobile nucleotides to broader training perturbations while keeping constrained regions

sharper, and the model learns to recover target conformers from these calibrated noisy frames. The bridge distribution therefore provides a source of diversity while retaining target-conformer supervision.

The input sequence s is encoded once by a frozen pretrained ALPHAFOLD3 (AF3) Pairformer [5] into single and pair representations $(\mathbf{c}^{(s)}, \mathbf{c}^{(p)})$. A FlexibilityNet ψ learns a sequence-derived flexibility condition grounded by B-factor-derived mobility proxies during training, while an IPA-based bridge-field network v_θ [12] predicts translational and rotational bridge velocities from the noisy state G_t, time t, AF3 representations, and this flexibility condition. The predicted fields are analytically inverted to target backbone frames and combined with torsions $\hat{\Phi} \in (S^1)^{10L}$ for all-atom reconstruction. At inference, REFLEX uses only the input sequence and does not require experimental flexibility annotations or re-computed AF3 features under perturbed evolutionary contexts.

3.1 Flexibility-calibrated Heteroscedastic Bridge

The heteroscedastic bridge trains the model to recover each target conformer from perturbed intermediate states. Small-noise analyses of supervised learning and denoising connect input corruption to derivative-based regularization, providing a rationale for improved local robustness [21, 22]. Motivated by this view, REFLEX exposes flexible residues to broader perturbations while keeping constrained regions sharper. We assess the resulting robustness in the recovery-stress experiments (Section 4.5); Appendix B gives a brief discussion.

To set these widths, we use experimental B-factors as within-chain mobility proxies. We therefore normalize B-factors within each chain:

$$\widetilde{B}_i = \frac{B_i}{\sum_{j=1}^{L} B_j / L}, \qquad i = 1, \ldots, L. \tag{4}$$

We rescale this proxy as $\alpha_i = (\mathrm{clip}(\widetilde{B}_i, 0.2, 1.6) - 0.2)/1.4 \in [0, 1]$ and set the variance multiplier $a_i = (1 + 2\alpha_i)^2$. The corresponding standard-deviation multiplier is $\sqrt{a_i} = 1 + 2\alpha_i$, consistent with Table 4. This calibration changes the training bridge only; no B-factor annotation is used at inference.

Translation bridge. For translations, the calibrated bridge width can be applied directly in $\mathbb{R}^3$. For a nucleotide with variance multiplier a, define $\gamma_x(t; a) = \sigma_x \sqrt{a\, t(1-t)}$, where σ_x is a global translational scale. The translational bridge uses the Gaussian marginal:

$$q_t^x(\cdot \mid x_0, x_1) = \mathcal{N}\left((1-t)x_0 + t x_1, \gamma_x^2(t; a) I_3\right). \tag{5}$$

This Gaussian marginal provides a Euclidean carrier for the calibrated uncertainty. The factor $t(1-t)$ makes the stochastic term vanish at the source and target structures, while the multiplier a determines how much intermediate uncertainty the model sees at this nucleotide during training. Thus, stochasticity is calibrated at the nucleotide level.

Rotation bridge. The same calibration principle is applied to nucleotide orientations. Since rotations lie on $\mathrm{SO}(3)$, we place the stochastic perturbation in a local logarithmic chart centered on the principal geodesic from R_0 to R_1. Let

$$\Omega = \log(R_0^\top R_1)^\vee, \qquad \bar{R}_t = R_0 \exp(t\Omega^\wedge)$$

be the geodesic center at time t. For the same variance multiplier a, define $\gamma_R(t; a) = \sigma_R \sqrt{a\, t(1-t)}$, where σ_R is a global rotational scale. The rotation bridge is the distribution induced by a Gaussian perturbation in the tangent coordinate around $\bar{R}_t$:

$$q_t^R(\cdot \mid R_0, R_1) = (\Phi_{\bar{R}_t})_{\#}\, \mathcal{N}\left(0, \gamma_R^2(t; a) I_3\right), \qquad \Phi_{\bar{R}_t}(\xi) = \bar{R}_t \exp(\xi^\wedge). \tag{6}$$

3.2 Learning the Flexibility Condition

FlexibilityNet learns an explicit sequence-derived flexibility condition. This condition is grounded by relative B-factor mobility during training and guides the bridge-field network during recovery. The input sequence s is encoded once by the frozen AF3 Pairformer into $(\mathbf{c}^{(s)}, \mathbf{c}^{(p)})$. FlexibilityNet ψ maps these representations to supervised flexibility logits and to the flexibility condition $h_\psi(s)$:

$$(o_\psi(s), h_\psi(s)) = \psi(\mathbf{c}^{(s)}, \mathbf{c}^{(p)}), \tag{7}$$

where $o_\psi(s) \in \mathbb{R}^{L \times K_B}$ parameterizes a K_B-bucket flexibility classifier and $h_\psi(s)$ is passed to the bridge-field network as an additional condition.

The supervised flexibility output provides a training signal: it encourages $h_\psi(s)$ to encode residue-wise mobility information derived from relative B-factors. The classifier head anchors this condition during training, and ensemble generation uses $h_\psi(s)$ to guide the bridge-field network.

Flexibility grounding. During training, the chain-normalized B-factor profile grounds the flexibility head. Let $\mathcal{V} \subseteq \mathcal{R}$ be the supervised nucleotide set with valid B-factor annotations. We discretize the chain-normalized proxy $\widetilde{B}_i$ into bucket labels $b_\mathcal{V}$ over $\mathcal{V}$, and train the logits with cross entropy:

$$\mathcal{L}_{\mathrm{B}} = \mathrm{CE}(o_{\psi,\mathcal{V}}(s), b_\mathcal{V}) \,. \tag{8}$$

This loss aligns the learned flexibility representation with relative mobility patterns observed in training structures. It does not provide test-time B-factors or manually specify flexible regions during sampling. If no valid annotations are available, we omit $\mathcal{L}_{\mathrm{B}}$ for that example.

3.3 Training Objective

The bridge construction in Section 3.1 gives noisy states $G_t = (x_t, R_t)$. The IPA-based network v_θ [12] receives G_t, t, $(\mathbf{c}^{(s)}, \mathbf{c}^{(p)})$, and $h_\psi(s)$, and predicts translation and rotation bridge fields $(\hat{u}_t^x, \hat{u}_t^R)$. We analytically map the predicted fields back to a target frame for supervision:

$$\hat{x}_1 = x_t + (1-t)\hat{u}_t^x, \qquad \hat{R}_1 = R_t \exp\big(((1-t)\hat{u}_t^R)^\wedge\big) \,. \tag{9}$$

This target-frame supervision encourages consistent recovery of the target under heteroscedastic bridge noise. It also couples the $h_\psi(s)$-conditioned bridge-field network to the heteroscedastic training bridge: flexible nucleotides are learned from broader intermediate perturbations, while rigid nucleotides are learned from sharper bridge neighborhoods. We define

$$\mathcal{L}_{\mathrm{SB}} = \mathbb{E}_{t,G_0,G_1}\left[\frac{1}{(1-t)^2}\left(\lambda_x \left\|\hat{x}_1 - x_1\right\|_{\mathbb{R}^3}^2 + \lambda_R \left\|\log_{R_t}(\hat{R}_1) - \log_{R_t}(R_1)\right\|_{\mathrm{SO}(3)}^2\right)\right] . \tag{10}$$

Here $\log_R(Q) = \log(R^\top Q)^\vee$, and the norms aggregate over resolved frame nucleotides. We use $\lambda_x = 2$ and $\lambda_R = 1$, as listed in Table 4.

The full training objective combines target-frame supervision with flexibility grounding and an auxiliary structural reconstruction loss:

$$\mathcal{L}_{\texttt{REFLEX}} = \mathcal{L}_{\mathrm{SB}} + \lambda_{\mathrm{B}}\,\mathcal{L}_{\mathrm{B}} + \lambda_{\mathrm{aux}}\,\mathcal{L}_{\mathrm{aux}}. \tag{11}$$

The auxiliary term $\mathcal{L}_{\mathrm{aux}}$, defined in Appendix F, supervises the reconstructed geometry from $(\hat{x}_1, \hat{R}_1, \hat{\Phi})$. Through this objective, B-factor information affects REFLEX only at training time: it calibrates the bridge distribution and grounds the flexibility representation. At inference, generation is driven by the bridge-field network conditioned on the sequence-derived embedding $h_\psi(s)$.

3.4 Inference

To generate the ensemble given an RNA sequence, we first compute the AF3 representations. During sampling, each sample k starts from an independent prior $G_0^{(k)} \sim p_0$. At each integration step, the conditional drift $\hat{u}_t^x, \hat{u}_t^R$ is predicted given the current frame state G_t, AF3 representations and the predicted flexibility condition. Given the predicted drift, the update is given by:

$$\begin{aligned} x_{t+\Delta t} &= x_t + \Delta t\, \hat{u}_t^x + g_x(t)\sqrt{\Delta t}\, \xi_t^x, \\ R_{t+\Delta t} &= R_t \exp\left(\left[\Delta t\, \hat{u}_t^R + g_R(t)\sqrt{\Delta t}\, \xi_t^R\right]^\wedge\right) . \end{aligned} \tag{12}$$

Here $\xi_t^x, \xi_t^R \sim \mathcal{N}(0, I_3)$ are independent standard Gaussian noise vectors. After the frame trajectory reaches $t = 1$, the predicted torsions $\hat{\Phi}$ and nucleotide-specific reference geometry reconstruct all heavy-atom coordinates. Repeating the procedure with independent initial states and sampler noise yields the generated ensemble.

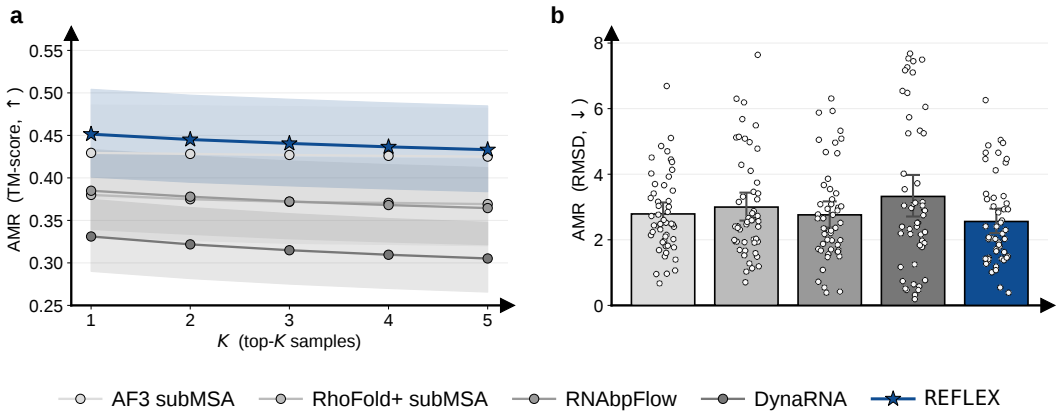


Figure 2: **Coverage of experimental conformers. a**: Top-K AMR (TM-score), the mean of the K highest matching scores for each reference conformer, for $K = 1, \ldots, 5$, evaluated on clusters where every method has at least five samples. **b**: Best-match RMSD AMR (Å) on the test set. Shading and error bars show 95% bootstrap confidence intervals; points in **b** show per-cluster values.

4 Experiments

We evaluate REFLEX around three questions. First, does the generated ensemble improve recovery of experimentally observed conformers? Second, do generated samples remain near observed states as ensemble diversity increases? Third, does the flexibility-calibrated bridge improve conformer-specific target-frame recovery under controlled perturbations? The first two questions evaluate final generated ensembles; the third isolates the bridge-field mechanism using the ablation variants. Metric definitions and statistical tests are given in Appendix C and Appendix E.

4.1 Dataset construction

Our RNA dataset totals 1,301 structures, curated from PDB [23], retaining chains of 20–300 nt with resolution ≤ 4.0 Å. We use a temporal cutoff of June 1, 2023 and an 80% CD-HIT sequence-identity exclusion criterion between training and test sequences. The test set contains 50 multi-conformer RNA clusters. REFLEX is trained on the corresponding pre-cutoff training split.

4.2 Baselines

We compare against two baseline families. The first uses single-structure predictors with inference-time MSA perturbations: AlphaFold3 [5] and RhoFold+ [6] with AFCluster-style MSA subsampling [7]. We disable templates and use the same fixed random seed for all subsampling runs. The second family consists of dedicated ensemble generators: RNAbpFlow [8], conditioned on sequence and MXFold2 [24] predicted base-pair information, and DynaRNA [9], a coordinate-space diffusion model initialized from a randomly selected experimental conformer of the target RNA. For ensemble evaluation, all methods use a common budget of $K = 20$ samples.

All external baselines use released models or checkpoints and are not retrained on our curated split. The benchmark therefore compares released systems under a common evaluation protocol; component-level attribution relies on the matched REFLEX-family ablations.

For the recovery-stress ablations, we compare REFLEX with two bridge variants trained under the same backbone architecture. Linear Bridge corresponds to the standard deterministic flow-matching path, while Uniform Stochastic Bridge uses a single global bridge width for all nucleotides and removes the flexibility head. All three models use the same recovery-stress protocol, detailed in Appendix C.

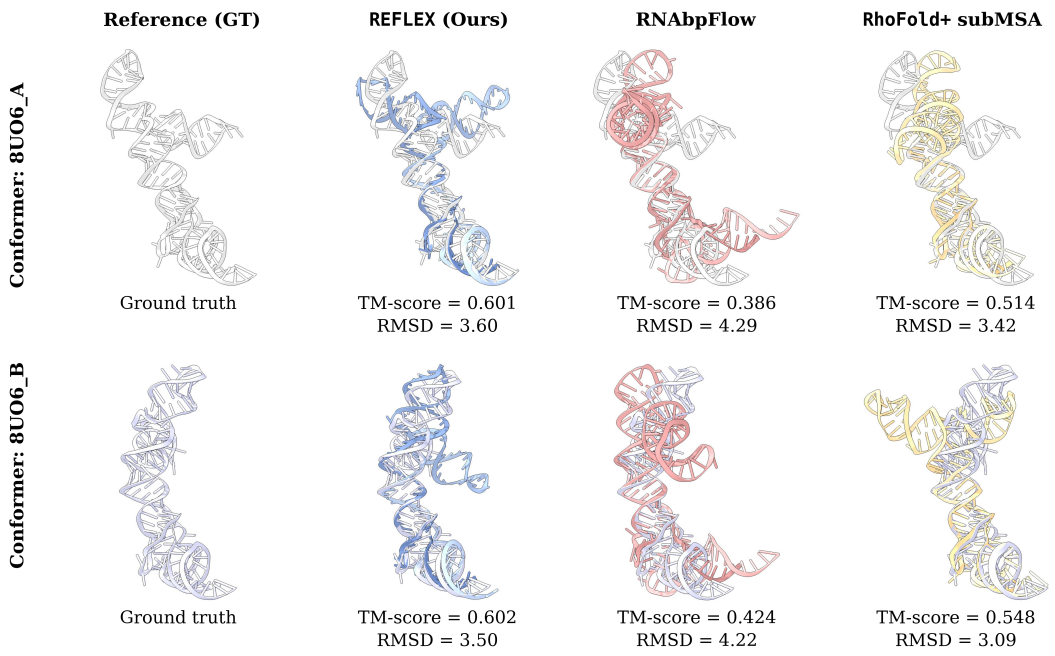


Figure 3: **Qualitative recovery on PDB** 8UO6**.** Two experimentally resolved conformers are shown against their nearest generated matches from REFLEX, RNABPFLOW, and RHOFOLD+ subMSA. TM-score and RMSD (Å) are reported below each match.

4.3 REFLEX improves recovery of observed conformers

Table 2 reports full-ensemble recovery with $K = 20$ samples per RNA. AMR-R averages each resolved conformer's best TM-score among generated samples; AMR-P averages each generated sample's best TM-score among resolved conformers. Both are computed per cluster and then macro-averaged. REFLEX achieves the highest AMR-R (0.451) and AMR-P (0.419) among the evaluated systems, compared with 0.430 and 0.411 for the strongest baseline, AF3 subMSA.

The paired AMR-R gain over AF3 subMSA is 0.022 (cluster-bootstrap 95% CI: $[0.006, 0.037]$), with higher recovery on 70% of test clusters and a Holm-adjusted one-sided Wilcoxon $p = 1.2 \times 10^{-3}$. These results support improved recovery on this benchmark.

Figure 3 illustrates conformer matching on PDB 8UO6, which contains two resolved conformations. This example provides a structure-level view of generated matches; evidence for performance across targets comes from the cluster-level benchmark and paired analysis above.

At a finer structural scale, we also assess whether recovered conformers preserve experimental base-pair interactions. For each reference conformer, we annotate its highest-TM-score generated match with DSSR [25] and compare its base pairs with those of the reference. Matching uses only TM-score, so base-pair agreement provides a complementary measure of secondary-structure recovery. Precision, recall, and F1 are macro-averaged across test clusters; evaluation details are given in Appendix H.

Table 1: **DSSR base-pair consistency of TM-score-selected matches.** Higher values are better; the best observed value in each column is in **bold**.

Method	Precision	Recall	F1
AF3 subMSA	0.697	**0.787**	0.729
RNABPFLOW	0.712	0.757	0.725
RHOFOLD+ subMSA	0.694	0.630	0.647
REFLEX	**0.734**	0.765	**0.741**

Table 2: **Ensemble recovery on the RNA benchmark.** All methods generate $K = 20$ samples per RNA. AMR-R and AMR-P report reference-to-sample and sample-to-reference best-match TM-scores. COV-R and COV-P use thresholds 0.30, 0.35, and 0.40. Each metric is macro-averaged over its evaluated clusters (Appendix C); higher AMR and coverage values are better, with the best in **bold**. Diversity is average pairwise RMSD in Å.

Method	AMR-R	COV-R			AMR-P	COV-P			Diversity (Å)
		@0.30	@0.35	@0.40		@0.30	@0.35	@0.40	
RNAbpFlow	0.385	0.640	0.422	0.358	0.378	0.558	0.386	0.371	3.178
AF3 subMSA	0.430	0.646	0.542	0.445	0.411	0.563	0.483	0.394	0.786
RhoFold+ subMSA	0.380	0.552	0.470	0.384	0.402	0.522	0.437	0.388	1.237
DynaRNA	0.331	0.436	0.327	0.247	0.257	0.250	0.218	0.204	4.439
REFLEX	**0.451**	**0.742**	**0.647**	**0.543**	**0.419**	**0.634**	**0.533**	**0.450**	2.900

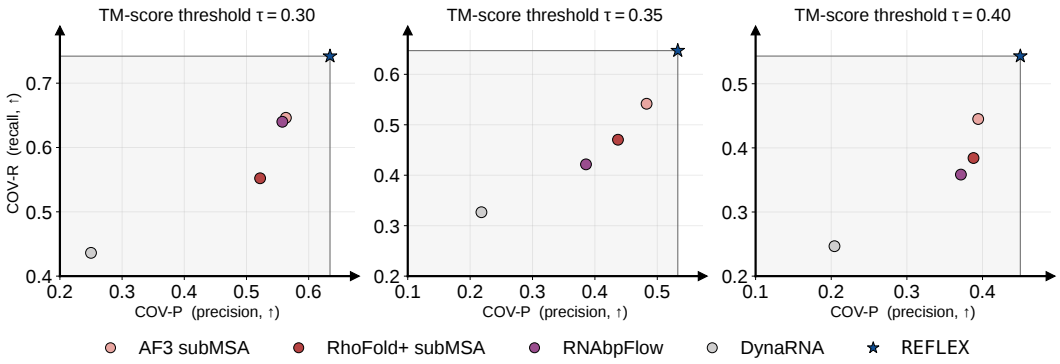


Figure 4: **Precision–recall Pareto plot.** Each method is evaluated at TM-score thresholds $\tau \in \{0.30, 0.35, 0.40\}$. COV-R and COV-P are macro-averaged as in Table 2; the gray shaded region denotes the dominated region.

Table 1 shows that REFLEX achieves the highest observed base-pair precision (0.734) and F1 (0.741), compared with 0.697 and 0.729 for AF3 subMSA. Its recall (0.765) exceeds that of RNAbpFlow and RhoFold+ subMSA, while AF3 subMSA attains the highest recall (0.787). Together with the global recovery metrics, these results indicate that REFLEX recovers conformational alternatives with competitive fidelity at the level of base-pair interactions. This evidence concerns the TM-score-selected matches and does not establish local validity across the entire generated ensemble.

4.4 REFLEX improves recovery while maintaining sample support

Table 2 evaluates whether generated ensembles place probability mass inside threshold-defined basins around experimentally observed conformers. COV-R measures how many observed conformers are recovered, while COV-P measures the complementary precision question: how many generated samples fall within one of these observed-state basins. REFLEX achieves the best COV-R and COV-P among the evaluated systems at every TM-score threshold $\tau \in \{0.30, 0.35, 0.40\}$. At $\tau = 0.40$, its mean per-cluster recovered-reference and matched-sample fractions are 54.3% and 45.0%, respectively. Recovery remains incomplete, so these gains do not establish exhaustive or near-native reconstruction of physical RNA ensembles.

The diversity results show that the coverage gains do not require the largest sample spread. REFLEX has average pairwise RMSD of 2.900 Å, above AF3 subMSA (0.786 Å) and RhoFold+ subMSA (1.237 Å), but below RNAbpFlow (3.178 Å) and DynaRNA (4.439 Å). Its higher COV-R and COV-P across all thresholds indicate a better balance of observed-conformer recovery and sample support on this benchmark.

4.5 REFLEX learns more reliable conformer-specific recovery

We examine conformer-specific recovery with Linear Bridge, Uniform Stochastic Bridge, and REFLEX. For each multi-conformer sequence, we perturb bridge states around each target conformer at $t \in \{0.25, 0.5, 0.75\}$ with corruption scale $\kappa \in \{0, 1, 2, 3\}$, then ask each model to recover the target in one step. Figure 5 shows the results at $t = 0.75$.

Figure 5 reports two complementary metrics. E_{frame} measures target-frame recovery error, so lower values indicate more accurate structural recovery; SepRet measures the ratio between recovered conformer separation and ground-truth conformer separation, so values closer to 1 indicate better preservation of conformer separation. At $t = 0.75$, REFLEX has lower mean recovery error than both alternatives at $\kappa = 2, 3$, with significant paired differences. At $\kappa = 0, 1$, the frame-error differences are not significant. Mean SepRet is closer to 1 for REFLEX across all tested corruption levels, with significant paired differences from both alternatives.

Table 3 stratifies the test set by ground-truth conformer distinctness. Uniform Stochastic Bridge has the highest mean AMR-R in the Low bin (0.540 versus 0.531 for REFLEX), while REFLEX leads in the Mid and High bins. The High-bin margin over Uniform Stochastic Bridge is only 0.002.

The recovery-stress analysis supports improved target-frame recovery under stronger perturbations, with gains that depend on bridge time and corruption strength. Matched ablations and FlexibilityNet validation are provided in Appendix H.

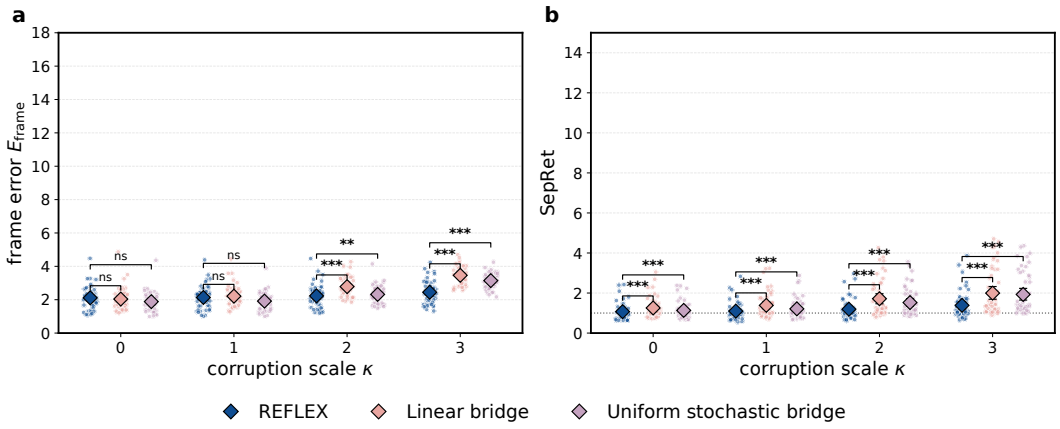


Figure 5: **Multi-conformer supervision stability.** **a**: Target-frame recovery error after one-step recovery from perturbed bridge states at $t = 0.75$. **b**: SepRet under the same perturbation protocol. Stars and ns denote significant and non-significant paired comparisons with REFLEX. Additional results at $t = 0.25$ and $t = 0.5$ are provided in Appendix D.

Table 3: **Conformer recovery by ground-truth distinctness.** Distinctness is average pairwise TM-distance between resolved conformers. The Low/Mid/High bins contain $16/17/17$ clusters. Entries are mean per-cluster AMR-R over the evaluated structure sets; higher is better.

Method	Low $[0.078, 0.363]$	Mid $[0.365, 0.471]$	High $[0.483, 0.843]$
Linear Bridge	0.505	0.435	0.340
Uniform Stochastic Bridge	**0.540**	0.444	0.356
REFLEX	0.531	**0.470**	**0.358**

5 Related Work

Traditional RNA structure prediction. Traditional RNA tertiary-structure modeling largely relied on secondary-structure-aware motif assembly, fragment assembly, and coarse-grained or energy-based search. MC-Fold/MC-Sym uses nucleotide cyclic motifs to connect sequence-level folding with

three-dimensional assembly [26], while RNAComposer and 3dRNA automate secondary-structure-guided construction from fragment or template libraries [27, 28]. Sampling-based methods such as SimRNA and FARFAR2 explore candidate folds with coarse-grained statistical potentials or Rosetta refinement [29, 30]. This line of work organizes RNA 3D prediction around a pipeline of structural constraints, candidate assembly or sampling, and subsequent scoring or refinement.

Learning-based RNA modeling. Recent learning-based predictors have substantially improved RNA structure prediction. trRosettaRNA, RhoFold+, NuFold, and AlphaFold3 predict accurate representative structures using learned sequence, evolutionary, or geometric representations [31, 6, 32, 5]. This family of methods shifts RNA structure prediction from hand-designed assembly and search toward learned structural priors trained on available RNA structures. Alternative conformations can be obtained by inference with heuristically subsampled MSAs, which probes conformational variation by changing the evolutionary context supplied to a structure predictor [7].

RNA inverse design and co-design have also been explored through reinforcement-learning-guided diffusion, joint topology-and-geometry objectives, and cooperative multi-agent learning [33–35]. These approaches address RNA design, complementing the sequence-conditioned conformational ensemble modeling studied here.

Flow matching models. Flow matching learns a continuous path that transports samples from a tractable prior to the data distribution [13–15, 36–39]. For molecular structures, geometric generative models define such dynamics on Euclidean coordinates, rotations, or $\mathrm{SE}(3)$ frames [40, 41, 20]. RNA-specific generative models have adapted these ideas to backbone-frame generation, base-pair-augmented backbone generation, and diffusion-based conformational modeling [12, 8, 9].

6 Conclusion

We presented REFLEX, a sequence-conditioned RNA ensemble generator that models conformational diversity as a nucleotide-wise property of the molecule. The method learns a flexibility-calibrated heteroscedastic stochastic bridge on $\mathrm{SE}(3)$, using chain-normalized experimental B-factors during training to set local stochastic strength and a sequence-derived flexibility condition at inference. On the test set, REFLEX better recovers experimentally observed conformational states, places more generated structures inside observed-state basins, and achieves the strongest precision–recall trade-off while maintaining meaningful diversity. The recovery-stress ablation indicates that target-frame recovery gains depend on bridge time and corruption strength: at $t = 0.75$, recovery error is significantly lower under stronger perturbations, while differences at lower corruption levels are not significant. Mean separation retention is closer to the reference value across the tested corruption levels. These results support the central claim that RNA ensemble generation benefits from calibrating where stochasticity is placed, providing a route toward sequence-conditioned models that capture both global alternative states and local flexibility patterns.

Acknowledgments and Disclosure of Funding

This work was supported by the UKRI Future Leaders Fellowship under grants MR/S017062/1 and MR/X011135/1; the National Natural Science Foundation of China under grants 62376056 and 62076056; Isambard AI Open Access (AIOA - 01 - 2026-06-04 - EvoRNA: Predictive Scaling of Evolution-Aware and Multimodal RNA Foundation Models for Crop Resilience); GPU compute credits provided through the Modal for Academics program; the Royal Society Faraday Discovery Fellowship (FDF/S2/251014); BBSRC Transformative Research Technologies (UKRI1875); the Royal Society International Exchanges Award (IES/R3/243136); the Kan Tong Po Fellowship (KTP/R1/231017); the Amazon Research Award; and the Alan Turing Fellowship.

References

[1] Anthony M Mustoe, Charles L Brooks, and Hashim M Al-Hashimi. Hierarchy of RNA functional dynamics. *Annual review of biochemistry*, 83(1):441–466, 2014.

[2] Laura R Ganser, Megan L Kelly, Daniel Herschlag, and Hashim M Al-Hashimi. The roles of structural dynamics in the cellular functions of RNAs. *Nature reviews Molecular cell biology*, 20(8):474–489, 2019.

[3] Maja Marušič, Judith Schlagnitweit, and Katja Petzold. RNA dynamics by NMR spectroscopy. *ChemBioChem*, 20(21):2685–2710, 2019.

[4] Megan L Ken, Rohit Roy, Ainan Geng, Laura R Ganser, Akanksha Manghrani, Bryan R Cullen, Ursula Schulze-Gahmen, Daniel Herschlag, and Hashim M Al-Hashimi. RNA conformational propensities determine cellular activity. *Nature*, 617(7962):835–841, 2023.

[5] Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J. Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with AlphaFold3. *Nature*, 630(8016):493–500, 2024.

[6] Tao Shen, Zhihang Hu, Siqi Sun, Di Liu, Felix Wong, Jiuming Wang, Jiayang Chen, Yixuan Wang, Liang Hong, Jin Xiao, Liangzhen Zheng, Tejas Krishnamoorthi, Irwin King, Sheng Wang, Peng Yin, James J. Collins, and Yu Li. Accurate RNA 3D structure prediction using a language model-based deep learning approach. *Nat. Methods*, 21(12):2287–2298, 2024.

[7] Hannah K. Wayment-Steele, Adedolapo Ojoawo, Renee Otten, Julia M. Apitz, Warintra Pitsawong, Marc Hömberger, Sergey Ovchinnikov, Lucy Colwell, and Dorothee Kern. Predicting multiple conformations via sequence clustering and AlphaFold2. *Nature*, 625(7996):832–839, 2024.

[8] Sumit Tarafder and Debswapna Bhattacharya. RNAbpFlow: Base pair-augmented SE(3)-flow matching for conditional RNA 3D structure generation. *bioRxiv*, 2025.

[9] Zhengxin Li, Junjie Zhu, Xiaokun Hong, Junxi Mu, Zhuoqi Zheng, Taeyoung Cui, Yutong Sun, Ting Wei, and Hai-Feng Chen. DynaRNA: accurate dynamic RNA conformation ensemble generation with diffusion model. *Communications Biology*, 8(1):1472, 2025.

[10] Oliviero Carugo. Atomic displacement parameters in structural biology. *Amino Acids*, 50(7):775–786, 2018.

[11] Ivantha Guruge, Ghazaleh Taherzadeh, Jian Zhan, Yaoqi Zhou, and Yuedong Yang. B-factor profile prediction for RNA flexibility using support vector machines. *Journal of Computational Chemistry*, 39(8): 407–411, 2018.

[12] Rishabh Anand, Chaitanya K. Joshi, Alex Morehead, Arian Rokkum Jamasb, Charles Harris, Simon V. Mathis, Kieran Didi, Rex Ying, Bryan Hooi, and Pietro Lio. RNA-FrameFlow: Flow matching for de novo 3D RNA backbone design. *Trans. Mach. Learn. Res.*, 2025, 2025.

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

[14] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

[15] Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *arXiv preprint arXiv:2302.00482*, 2023.

[16] Zhoutong Sun, Qian Liu, Ge Qu, Yan Feng, and Manfred T Reetz. Utility of B-factors in protein science: interpreting rigidity, flexibility, and internal motion and engineering thermostability. *Chemical reviews*, 119(3):1626–1665, 2019.

[17] Yunfei Hu, Kai Cheng, Lichun He, Xu Zhang, Bin Jiang, Ling Jiang, Conggang Li, Guan Wang, Yunhuang Yang, and Maili Liu. NMR-based methods for protein analysis. *Analytical chemistry*, 93(4):1866–1879, 2021.

[18] Agustí Emperador, Tim Meyer, and Modesto Orozco. Protein flexibility from discrete molecular dynamics simulations using quasi-physical potentials. *Proteins: Structure, Function, and Bioinformatics*, 78(1): 83–94, 2010.

[19] Petr Kouba, Joan Planas-Iglesias, Jiri Damborsky, Jiri Sedlar, Stanislav Mazurenko, and Josef Sivic. Learning to engineer protein flexibility. *arXiv preprint arXiv:2412.18275*, 2024.

[20] Jason Yim, Andrew Campbell, Andrew Y. K. Foong, Michael Gastegger, José Jiménez-Luna, Sarah Lewis, Victor Garcia Satorras, Bastiaan S. Veeling, Regina Barzilay, Tommi Jaakkola, and Frank Noé. Fast protein backbone generation with SE(3) flow matching. In *NeurIPS 2023 Machine Learning in Structural Biology Workshop*, 2023.

[21] Christopher M. Bishop. Training with noise is equivalent to Tikhonov regularization. *Neural Computation*, 7(1):108–116, 1995. doi: 10.1162/neco.1995.7.1.108.

[22] Guillaume Alain and Yoshua Bengio. What regularized auto-encoders learn from the data-generating distribution. *Journal of Machine Learning Research*, 15(110):3743–3773, 2014.

[23] Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic acids research*, 28(1):235–242, 2000.

[24] Kengo Sato, Manato Akiyama, and Yasubumi Sakakibara. RNA secondary structure prediction using deep learning with thermodynamic integration. *Nature communications*, 12(1):941, 2021.

[25] Xiang-Jun Lu, Harmen J. Bussemaker, and Wilma K. Olson. DSSR: an integrated software tool for dissecting the spatial structure of RNA. *Nucleic Acids Research*, 43(21):e142, 2015. doi: 10.1093/nar/gkv716.

[26] Marc Parisien and François Major. The MC-Fold and MC-Sym pipeline infers RNA structure from sequence data. *Nature*, 452(7183):51–55, 2008. doi: 10.1038/nature06684.

[27] Mariusz Popenda, Marta Szachniuk, Maciej Antczak, Katarzyna J. Purzycka, Piotr Lukasiak, Natalia Bartol, Jacek Blazewicz, and Ryszard W. Adamiak. Automated 3D structure composition for large RNAs. *Nucleic Acids Research*, 40(14):e112–e112, 2012. doi: 10.1093/nar/gks339.

[28] Yunjie Zhao, Yangyu Huang, Zhou Gong, Yanjie Wang, Jianfen Man, and Yi Xiao. Automated and fast building of three-dimensional RNA structures. *Scientific Reports*, 2(1), 2012. doi: 10.1038/srep00734.

[29] Michal J. Boniecki, Grzegorz Lach, Wayne K. Dawson, Konrad Tomala, Pawel Lukasz, Tomasz Soltysinski, Kristian M. Rother, and Janusz M. Bujnicki. SimRNA: a coarse-grained method for RNA folding simulations and 3D structure prediction. *Nucleic Acids Research*, 44(7):e63, 2016. doi: 10.1093/nar/gkv1479.

[30] Andrew Martin Watkins, Ramya Rangan, and Rhiju Das. FARFAR2: Improved de novo Rosetta prediction of complex global RNA folds. *Structure*, 28(8):963–976.e6, 2020. doi: 10.1016/j.str.2020.05.011.

[31] Wenkai Wang, Chenjie Feng, Renmin Han, Ziyi Wang, Lisha Ye, Zongyang Du, Hong Wei, Fa Zhang, Zhenling Peng, and Jianyi Yang. trRosettaRNA: automated prediction of RNA 3D structure with transformer network. *Nature Communications*, 14(1), 2023. doi: 10.1038/s41467-023-42528-4.

[32] Yuki Kagaya, Zicong Zhang, Nabil Ibtehaz, Xiao Wang, Tsukasa Nakamura, Pranav Deep Punuru, and Daisuke Kihara. NuFold: end-to-end approach for RNA tertiary structure prediction with flexible nucleobase center representation. *Nature communications*, 16(1):881, 2025.

[33] Tianmeng Hu, Yongzheng Cui, Biao Luo, and Ke Li. RIDER: 3D RNA inverse design with reinforcement learning-guided diffusion. In *Proc. International Conference on Learning Representations*, volume 2026, pages 79321–79346, 2026.

[34] Tianmeng Hu, Biao Luo, and Ke Li. TANGO: RNA topology and geometry co-design. *arXiv preprint arXiv:2610.20277*, 2026.

[35] Tianmeng Hu, Biao Luo, and Ke Li. CocoRNA: Collective RNA design with cooperative multi-agent reinforcement learning. In *Proc. International Conference on Machine Learning*, volume 306, pages 45495–45519, 2026.

[36] Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26(209):1–80, 2025.

[37] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[38] Yuyang Shi, Valentin De Bortoli, Andrew Campbell, and Arnaud Doucet. Diffusion Schrödinger bridge matching. *Advances in neural information processing systems*, 36:62183–62223, 2023.

[39] Alexander Tong, Nikolay Malkin, Kilian Fatras, Lazar Atanackovic, Yanlei Zhang, Guillaume Huguet, Guy Wolf, and Yoshua Bengio. Simulation-free Schrödinger bridges via score and flow matching. *arXiv preprint arXiv:2307.03672*, 2023.

[40] Jason Yim, Brian L Trippe, Valentin De Bortoli, Emile Mathieu, Arnaud Doucet, Regina Barzilay, and Tommi Jaakkola. SE(3) diffusion model with application to protein backbone generation. In *International Conference on Machine Learning (ICML)*, 2023.

[41] Avishek Joey Bose, Tara Akhound-Sadegh, Guillaume Huguet, Kilian Fatras, Jarrid Rector-Brooks, Cheng-Hao Liu, Andrei Cristian Nica, Maksym Korablyov, Michael Bronstein, and Alexander Tong. SE(3)-stochastic flow matching for protein backbone generation. *arXiv preprint arXiv:2310.02391*, 2023.

[42] Jack Cole, Heng Yang, Krasimira Tsaneva-Atanasova, and Ke Li. PhyloAug: An evolutionary and structure-aware data augmentation tool. *bioRxiv*, pages 2025–09, 2025.

[43] Shiting Wang, Yingjie Zou, and Ke Li. Separating common and unique directions for model merging, 2026.

[44] Yuan Li, Heng Yang, Renzhi Chen, and Ke Li. Understanding transfer learning of RNA foundation models on downstream tasks. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306, pages 71724–71744. PMLR, 2026.

[45] Shasha Zhou, Mingyu Huang, Jack Cole, Charles Britton, Ming Yin, Jan Wolber, and Ke Li. Assessing automated fact-checking for medical LLM responses with knowledge graphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(46):39646–39656, 2026. doi: 10.1609/aaai.v40i46.41317.

[46] Shasha Zhou, Mingyu Huang, and Ke Li. Position: Genomic model research must move beyond anecdotal evaluation of interpretability methods. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306, pages 172974–172999. PMLR, 2026.

A Hyperparameter Settings

REFLEX was trained on a single NVIDIA GH200 GPU. Table 4 summarizes the main hyperparameters.

Table 4: Key hyperparameters of REFLEX.

Group	Symbol / Name	Value
Time grid	Cutoff ϵ_t	10^{-2}
	Inference / validation steps	60 / 60
Stochastic bridge	Base widths σ_x / σ_R	1.0 Å / 1.0 rad
	Heteroscedastic gain	$\sigma_i = \sigma(1 + 2\alpha_i)$, with $\alpha_i = (\mathrm{clip}(\widetilde{B}_i, 0.2, 1.6) - 0.2)/1.4$
	Diffusion g_x / g_R	1.0 Å / 1.0 rad; constant
	Rotation schedule	Exponential, $\rho = 10$
FlexibilityNet	AF3 single input / hidden dimension	384 / 64
	Transformer layers / heads / FFN	2 / 4 / 256
	Flexibility bins K_B	64 over $\widetilde{B} \in [0.2, 1.6]$
	Condition $h_\psi(s)$	Bin probabilities; FiLM on single embedding
	Training status	Jointly trained with $\mathcal{L}_{\mathrm{B}}$
Flow model	Node / edge embedding c_s / c_z	256 / 128
	IPA blocks / heads / hidden dimension	4 / 8 / 128
	IPA query–key / value points	8 / 12
	Sequence transformer heads / layers	4 / 2
	Predicted torsions	10 (8 backbone + O2′ + χ)
Flow-branch loss	Translation / rotation weight	2.0 / 1.0
	Auxiliary weight / time threshold	1.0 / 0.25
	Torsion loss weight	1.0
Optimization	Optimizer	AdamW; constant learning rate
	Learning rate / weight decay	10^{-4} / 10^{-2}
	Batch size	24
	Self-conditioning probability	0.5
	Gradient clipping	1.0, global L_2 norm
	Training length	400 epochs / 22,000 steps

B Noise Calibration and Robust Recovery

Small-noise training can regularize local sensitivity [21, 22]. Under the regularity assumptions of Alain and Bengio [22], a Euclidean denoiser r_η with $r_\eta(x) = x + o(1)$ as $\eta \to 0$ satisfies

$$\begin{aligned}\mathcal{L}_{\mathrm{denoise}}(\eta) &= \mathbb{E}_{x,z}\big\|r_\eta(x + \eta z) - x\big\|_2^2 \\ &= \mathbb{E}_x\big\|r_\eta(x) - x\big\|_2^2 + \eta^2 \mathbb{E}_x\big\|J_{r_\eta}(x)\big\|_F^2 + o(\eta^2).\end{aligned}$$

Here x is a clean input, $z \sim \mathcal{N}(0, I)$ is independent noise, and J_{r_η} is the denoiser's Jacobian. The η^2-weighted term penalizes local sensitivity to input perturbations.

This denoising result motivates our bridge design. In REFLEX, varying bridge noise keeps the target conformer fixed, encouraging recovery over a local neighborhood. Flexibility calibration widens this neighborhood at mobile residues while keeping constrained regions sharper. We use bounded noise scales to retain target-specific information, and assess robustness empirically in Section 4.5.

C Ensemble Evaluation Metrics

For a cluster with K generated samples and K_c experimentally resolved states, let $\mathcal{G} = \{g_j\}_{j=1}^{K}$ denote the generated ensemble and $\mathcal{Y} = \{y_k\}_{k=1}^{K_c}$ the experimental ensemble. These metrics test three desiderata for ensemble generation: recovering experimentally observed conformational states, avoiding unsupported generated structures, and maintaining non-collapsed ensemble spread.

Fidelity. We measure recovery of experimentally observed states using average matched recovery. For each reference conformer y, order the TM-scores of all K generated samples as $s_{(1)}(y) \geq \cdots \geq s_{(K)}(y)$. Figure 2a reports top-$k$ AMR for $k = 1, \ldots, 5$:

$$\mathrm{AMR}_{\mathrm{TM}}@k \;=\; \frac{1}{K_c} \sum_{y \in \mathcal{Y}} \frac{1}{k} \sum_{r=1}^{k} s_{(r)}(y) \quad \uparrow .$$

Here k counts the highest-scoring matches averaged within the fixed generated ensemble. The best-match case gives $\mathrm{AMR\text{-}R} = \mathrm{AMR}_{\mathrm{TM}}@1$, as reported in Table 2. Figure 2b reports best-match RMSD AMR over the same generated ensemble:

$$\mathrm{AMR}_{\mathrm{RMSD}} \;=\; \frac{1}{K_c} \sum_{y \in \mathcal{Y}} \min_{g \in \mathcal{G}} \mathrm{RMSD}(g,\, y) \quad \downarrow .$$

The complementary sample-to-reference score is

$$\mathrm{AMR\text{-}P} \;=\; \frac{1}{K} \sum_{g \in \mathcal{G}} \max_{y \in \mathcal{Y}} \mathrm{TM}(g,\, y) \quad \uparrow .$$

We evaluate AMR-R, AMR-P, COV-R, and COV-P on the RNA benchmark. Each metric is computed per cluster and macro-averaged over its evaluated clusters.

Precision–recall trade-off. At TM-score threshold τ, recall coverage measures how many experimental states are covered by the generated ensemble, while precision coverage measures how many generated samples remain near the experimental ensemble:

$$\mathrm{COV\text{-}R}_{\tau} \;=\; \frac{1}{K_c} \sum_{k=1}^{K_c} \mathbf{1}\left[\max_{g \in \mathcal{G}} \mathrm{TM}(g,\, y_k) \;\geq\; \tau\right] \quad \uparrow,$$

$$\mathrm{COV\text{-}P}_{\tau} \;=\; \frac{1}{K} \sum_{j=1}^{K} \mathbf{1}\left[\max_{y \in \mathcal{Y}} \mathrm{TM}(g_j,\, y) \;\geq\; \tau\right] \quad \uparrow .$$

Diversity. We quantify ensemble spread as the average pairwise RMSD among generated samples:

$$\mathrm{diversity}(\mathcal{G}) \;=\; \binom{K}{2}^{-1} \sum_{1 \leq i < j \leq K} \mathrm{RMSD}(g_i,\, g_j) \quad \uparrow .$$

AMR and COV-R measure recovery of the experimental states, COV-P measures generated-sample fidelity to the experimental ensemble, and pairwise RMSD is used descriptively. We do not treat diversity alone as sufficient evidence of ensemble quality.

Ground-truth conformer distinctness. For the bridge ablation, we stratify test clusters by how separated their experimentally resolved conformers are. We measure this separation by average pairwise TM-distance,

$$D_{\mathrm{GT}}(\mathcal{Y}) \;=\; \binom{K_c}{2}^{-1} \sum_{1 \leq i < j \leq K_c} (1 - \mathrm{TM}(y_i,\, y_j)) \quad \uparrow,$$

where larger values indicate more distinct observed conformers.

Conformer-specific recovery stability. The recovery-stress analysis uses $K = 150$ samples per ablation variant and aggregates results over paired evaluation clusters. We test whether the learned field recovers each supervised conformer from nearby off-path bridge states. Let $G_1^{(m)}$ denote an experimentally observed conformer of a sequence, and let $G_t^{(m,\kappa)}$ denote a bridge state around that conformer at time t with perturbation scale κ. Applying the predicted bridge field and the same target-frame recovery used in training gives a recovered conformer $\hat{G}_1^{(m,\kappa)}$. We report the target-frame recovery error

$$E_{\text{frame}}(\kappa) \;=\; \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} d_{\text{frame}}\Big(\hat{G}_1^{(m,\kappa)},\, G_1^{(m)}\Big) \quad \downarrow,$$

where d_{frame} is the frame-level distance used for the ablation. To test whether this recovery preserves distinct conformational states, we compute the separation retention ratio for conformer pairs,

$$\text{SepRet}_{a,b}(\kappa) \;=\; \frac{d_{\text{frame}}\Big(\hat{G}_1^{(a,\kappa)},\, \hat{G}_1^{(b,\kappa)}\Big)}{d_{\text{frame}}\Big(G_1^{(a)},\, G_1^{(b)}\Big)}.$$

SepRet values closer to 1 indicate that the recovered conformers preserve the ground-truth separation between conformers. Values much below 1 indicate collapse toward a shared structure, while values much above 1 indicate over-dispersion. We average SepRet over conformer pairs and clusters at each κ.

D Additional Results

We report additional recovery-stress results at $t = 0.25$ and $t = 0.5$ to complement Figure 5. At $t = 0.25$, frame-error improvements over Linear Bridge are significant at all tested corruption levels, while improvements over Uniform Stochastic Bridge are significant only at $\kappa = 3$. At $t = 0.5$, frame-error differences are not significant at $\kappa = 0, 1$; at $\kappa = 2$, only the comparison with Linear Bridge is significant, and at $\kappa = 3$, both comparisons are significant. At both bridge times, mean SepRet is closer to 1 for `REFLEX` across the tested corruption levels, with significant paired differences from both alternatives.

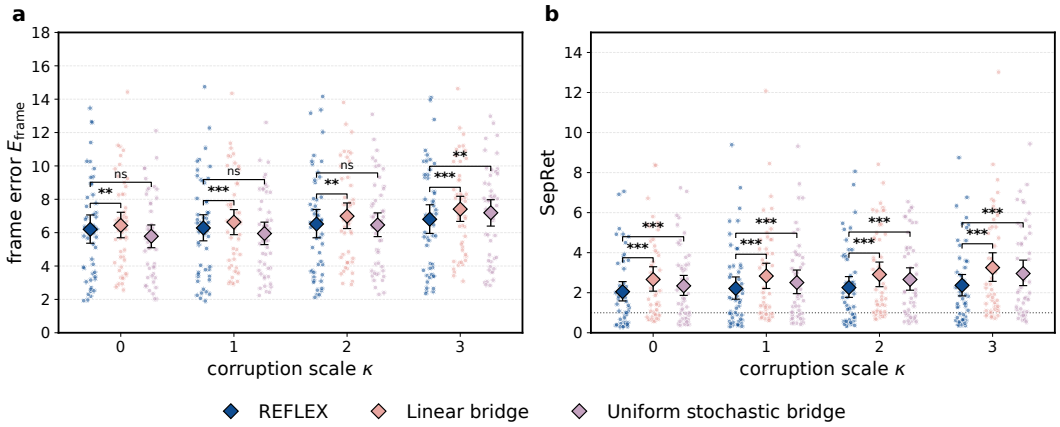


Figure 6: **Additional recovery-stress results at $t = 0.25$. a**: Target-frame recovery error after one-step recovery from perturbed bridge states. **b**: SepRet under the same perturbation protocol. Stars and `ns` denote significant and non-significant paired comparisons with `REFLEX`.

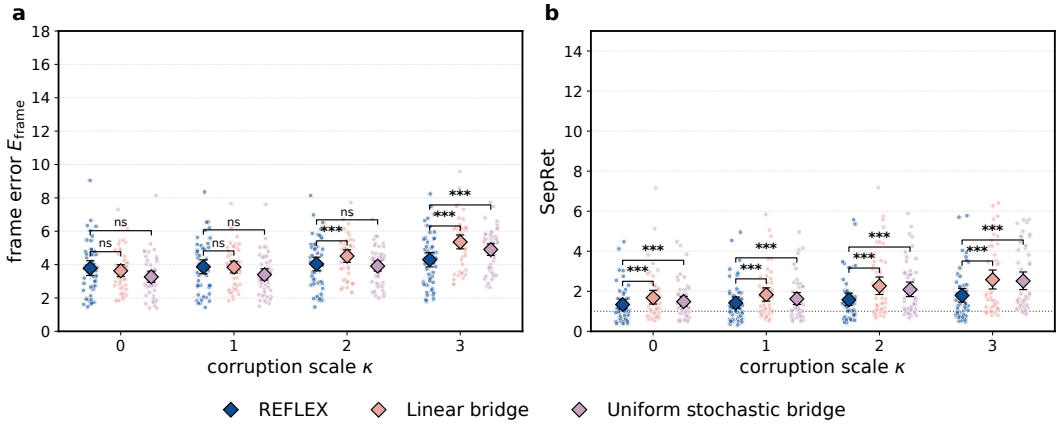


Figure 7: **Additional recovery-stress results at $t = 0.5$. a**: Target-frame recovery error after one-step recovery from perturbed bridge states. **b**: SepRet under the same perturbation protocol. Stars and `ns` denote significant and non-significant paired comparisons with `REFLEX`.

E Statistical Significance Analysis

For the AMR and coverage analyses, we compute each metric per test cluster and report cluster-level means with 95% bootstrap confidence intervals from $10{,}000$ resamples. Significance is assessed against `REFLEX` on the paired cluster intersection. For continuous AMR metrics, we use a one-sided Wilcoxon signed-rank test with the alternative that `REFLEX` is better. For thresholded COV-R and COV-P, we use a one-sided paired bootstrap on the mean cluster-wise difference $\Delta = \mathrm{REFLEX} - \mathrm{baseline}$. Within each metric, p-values across baseline comparisons are corrected with Holm's step-down procedure; stars in Figures 8 and 9 denote Holm-corrected significance.

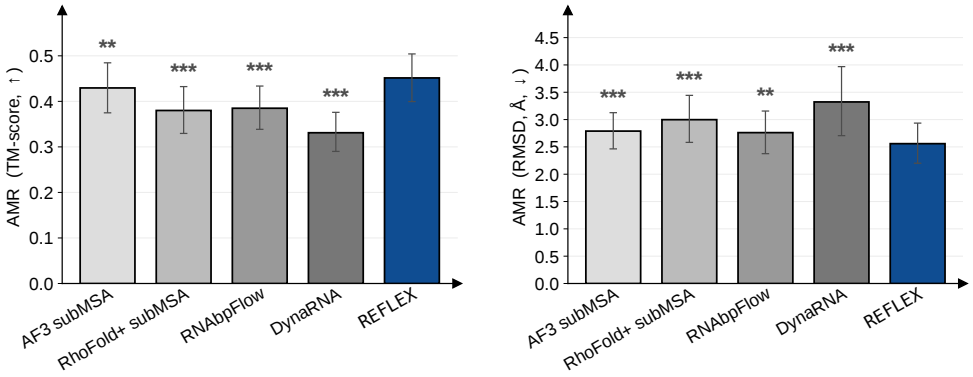


Figure 8: **Statistical analysis of conformer recovery.** Left: best-match TM-score AMR, where higher is better. Right: best-match RMSD AMR, where lower is better. Bars show means with 95% bootstrap confidence intervals; stars denote Holm-corrected significance from one-sided paired Wilcoxon comparisons with `REFLEX`.

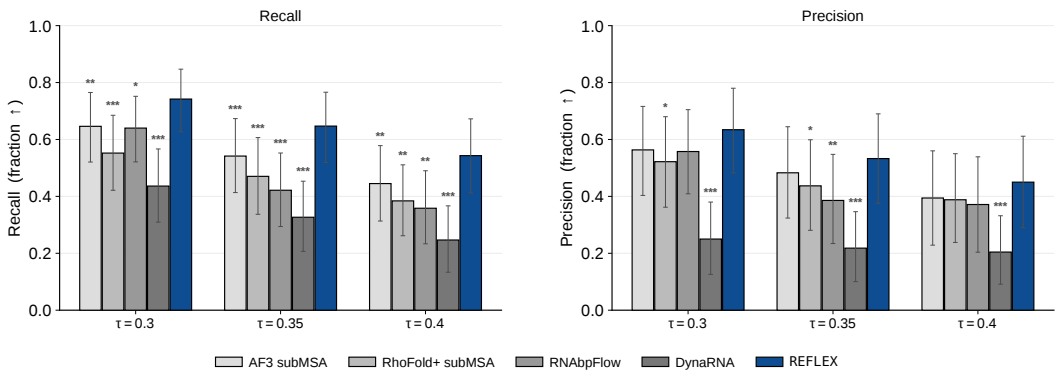


Figure 9: **Statistical analysis for precision–recall coverage.** Left: COV-R at TM-score thresholds $\tau \in \{0.30, 0.35, 0.40\}$. Right: COV-P at the same thresholds. Bars show cluster-level means with 95% bootstrap confidence intervals; stars denote Holm-corrected significance from one-sided paired bootstrap comparisons with REFLEX. Absence of a star indicates that the comparison does not meet the 0.05 significance threshold.

F All-atom Reconstruction and Auxiliary Loss

We extend the frame-based reconstruction of RNA-FRAMEFLOW [12] with ten torsions per nucleotide: eight backbone angles, one for $O2'$ placement, and the glycosidic angle χ. Nucleotide-specific reference geometry supplies complete bases, yielding all heavy-atom coordinates:

$$\hat{Y} = \mathrm{AllAtom}(\hat{x}_1, \hat{R}_1, \hat{\Phi}; s), \qquad Y = \mathrm{AllAtom}(x_1, R_1, \Phi; s). \tag{13}$$

The auxiliary geometry losses are

$$\mathcal{L}_{\mathrm{atom}} = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \|\hat{Y}_a - Y_a\|_2^2, \tag{14}$$

$$\mathcal{L}_{\mathrm{dist}} = \frac{1}{|\mathcal{P}|} \sum_{(a,b) \in \mathcal{P}} \Big(\|\hat{Y}_a - \hat{Y}_b\|_2 - \|Y_a - Y_b\|_2 \Big)^2, \tag{15}$$

$$\mathcal{L}_{\mathrm{tors}} = \frac{1}{|\mathcal{T}|} \big\| \hat{z}_{\mathcal{T}} - z_{\mathcal{T}} \big\|_2^2, \qquad z = (\cos\Phi, \sin\Phi),\ \hat{z} = (\cos\hat{\Phi}, \sin\hat{\Phi}). \tag{16}$$

These terms can be combined as

$$\ell_{\mathrm{target}} = \lambda_{\mathrm{atom}} \mathcal{L}_{\mathrm{atom}} + \lambda_{\mathrm{dist}} \mathcal{L}_{\mathrm{dist}} + \lambda_{\mathrm{tors}} \mathcal{L}_{\mathrm{tors}}. \tag{17}$$

We use this reconstruction term as $\mathcal{L}_{\mathrm{aux}}$ in Eq. (11), while Eq. (10) gives the frame-level stochastic bridge term.

G Limitations and Broader Impact

REFLEX uses chain-normalized experimental B-factors as weak supervision for relative local mobility during training, so its calibration depends on the availability and quality of experimentally resolved RNA structures. B-factors are affected by experimental modality, refinement, resolution, and static disorder, and should not be interpreted as absolute thermodynamic fluctuations. The evaluation is also limited by the scarcity of RNA sequences with multiple experimentally resolved conformers; observed structures may cover only part of the accessible ensemble and do not provide equilibrium populations. Finally, REFLEX focuses on sequence-conditioned RNA backbone-frame ensembles, and does not explicitly model ligand-bound switching, protein–RNA complexes, ion effects, or experimentally validated conformational populations.

Future work could explore adapting evolutionary and structure-aware data augmentation tools such as PhyloAug [42] to alleviate the scarcity of multi-conformer RNA training data, with validation of whether augmented examples preserve relevant conformational diversity. Integrating data from

complementary modalities, including sequence, structure, and experimental measurements, could further constrain conformational ensembles. Inspired by the separation of common and unique directions in model merging [43], a promising direction is to explore how multimodal integration can retain both shared information and modality-specific signals. Future work could also investigate task-aligned adaptation of pretrained RNA representations for conformational ensemble modeling, motivated by findings on the task-dependent transferability of RNA foundation models [44].

More broadly, REFLEX may help characterize RNA conformational heterogeneity and support hypothesis generation in settings where experimentally resolved ensembles are scarce. However, generated conformers should not be interpreted as experimentally validated structures or functional states. Related concerns motivate knowledge-graph-grounded factuality assessment of medical LLM responses [45] and systematic evaluation of consistency, faithfulness, and biological validity in genomic interpretability [46]. These perspectives motivate grounding downstream claims in independent evidence and systematically evaluating any mechanistic interpretations derived from generated ensembles. Downstream use may reflect biases in available RNA structures, pretrained representations, and the experimental proxies used for flexibility supervision. Care should therefore be taken when applying the method to biological design, functional interpretation, or therapeutic settings, where generated candidates require empirical validation.

H Additional Ablations and Validation

Matched ablations on ensemble generation. We evaluate matched ablations with $K = 20$ samples per RNA (Table 5). All variants share frozen AF3 representations, generator backbone, training data, and evaluation protocol. In addition to Linear Bridge and Uniform Stochastic Bridge, we remove flexibility supervision by setting $\lambda_{\mathrm{B}} = 0$ while retaining residue-dependent bridge widths, or remove per-residue calibration by using homoscedastic bridge noise while retaining $\mathcal{L}_{\mathrm{B}}$.

Table 5: **Matched component ablations.** All variants use $K = 20$ samples per RNA. Coverage is macro-averaged across clusters, with the best value in each column in **bold**. Diversity is average pairwise RMSD in Å. The two "w/o" rows each remove one component from REFLEX.

Variant	COV-R			COV-P			Diversity (Å)
	@0.30	@0.35	@0.40	@0.30	@0.35	@0.40	
Linear Bridge	0.665	0.570	0.449	0.554	0.427	0.383	3.401
Uniform Stochastic Bridge	0.687	0.601	0.434	0.606	0.466	0.404	3.088
w/o flexibility supervision	**0.753**	0.616	0.452	**0.647**	0.506	0.410	3.157
w/o per-residue calibration	0.717	0.620	0.436	0.620	0.471	0.401	3.252
REFLEX	0.742	**0.647**	**0.543**	0.634	**0.533**	**0.450**	2.900

REFLEX improves all six coverage metrics over Linear Bridge and Uniform Stochastic Bridge while producing lower diversity. Removing per-residue calibration reduces coverage at every threshold. Removing flexibility supervision slightly improves coverage at $\tau = 0.30$, but reduces COV-R/COV-P at $\tau = 0.40$ from $0.543/0.450$ to $0.452/0.410$. Thus, residue-dependent calibration provides consistent coverage gains, while flexibility supervision mainly benefits recovery at stricter thresholds.

FlexibilityNet prediction. We evaluate the 64-bin supervised prediction head $o_\psi(s)$ on 50 sequence-disjoint RNA chains with 5,857 annotated nucleotides. Targets are the chain-normalized B-factor-derived proxy discretized into 64 bins. We report accuracy within ± 2 bins and AUROC for identifying the top 25% of nucleotides within each chain according to this proxy. Metrics are computed per chain and macro-averaged; 95% confidence intervals use 10,000 bootstrap resamples of evaluation chains.

Table 6 indicates that the supervised head captures informative patterns in the relative proxy. These measurements do not establish agreement with absolute mobility, coordinate RMSF, or thermodynamic fluctuations. At inference, only the latent condition $h_\psi(s)$ is passed to the bridge-field network; the scalar predictions from the supervised head are not consumed by the sampler. The downstream contribution of flexibility grounding is assessed separately by the $\lambda_{\mathrm{B}} = 0$ ablation in Table 5.

Table 6: **FlexibilityNet prediction.** Metrics assess agreement with the relative B-factor-derived proxy.

Metric	Chain-macro result	95% CI
Accuracy within ± 2 bins	0.759	$[0.691, 0.824]$
AUROC for the top 25% by the proxy	0.681	$[0.630, 0.731]$

Base-pair evaluation protocol. For each experimental conformer, we select the highest-TM-score generated sample, as in AMR-R, without using base-pair information. For REFLEX, these structures include complete nucleobases reconstructed as in Appendix F. We annotate reference and matched structures with DSSR [25]. Base-pair precision, recall, and F1 are computed per match, averaged within each cluster, and then macro-averaged across test clusters.

Results are reported in Table 1 in Section 4.3. DynaRNA is excluded because its native output contains only C4′ coordinates. DSSR evaluation would require a separate full-atom reconstruction model, introducing dependence on additional post-processing.

I Fidelity on RNAs with a Single Deposited Structure

To assess best-sample fidelity beyond the multi-conformer benchmark, we evaluate 24 RNA chains (71–297 nt) with exactly one deposited experimental structure, using the same temporal cutoff and sequence-identity filtering against the training set. A single deposited conformation does not imply an intrinsically single-state RNA. All methods use the same settings as in the main ensemble benchmark. Experimental structures are used only for evaluation. Predictions are aligned with US-align on C4′ atoms (`-mol RNA`), with TM-score normalized by reference length and RMSD computed over aligned atom pairs.

Table 7: **Best-sample fidelity on 24 RNAs with a single deposited structure.** Values are means of per-RNA maximum TM-score and minimum RMSD over US-align-aligned C4′ atoms. The best observed values are in **bold**.

Method	Best TM-score ↑	Best RMSD (Å) ↓
AF3 subMSA	0.435	3.39
RhoFold+ subMSA	0.429	3.81
RNAbpFlow	0.425	3.01
REFLEX	**0.456**	**2.88**

REFLEX achieves the highest observed mean best TM-score (0.456) and lowest mean best RMSD (2.88 Å) among the four methods (Table 7), suggesting that ensemble generation remains compatible with competitive best-sample fidelity on these targets.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims and contributions are supported with experimental results in section 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations are discussed in Appendix G.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Proofs are in the Appendix B.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The method is described in Section 3 and implementation details are given in the Appendix A.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code will be released upon publication.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Experimental details are in the Section 4.1 and in the Appendix A.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report the statistical analysis in Appendix E and Section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Details are provided in the Appendix A.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the broader impacts of our work in Appendix G.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Code (AlphaFold3, RNA-FrameFlow), data (PDB) and software (ChimeraX) are properly credited and cited.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.