Secondary-Structure Reward Shaping for RL-Based RNA Inverse Design

Amanda New
NSERC USRA, Trinity Western University
Supervisor: Prof. Herbert Tsang

Working draft August 28, 2026

Abstract

RIDER fine-tunes a diffusion-based RNA sequence generator using a folding oracle and a tertiary-structure reward based on root-mean-square deviation (RMSD) and global distance test total score (GDT-TS). Its RhoFold oracle also produces a secondary-structure base-pairing prediction on each forward pass, but this output is not used by the original reward function. We introduce a secondary-structure reward based on base-pair F1 score and combine it with RIDER's existing tertiary reward. Because base-pair F1 is frequently exactly zero on sampled sequences, this term can provide little differentiation among trajectories in a sparse-signal regime. We therefore add a capped correct-pair shaping bonus that amplifies rare partial successes once they occur and scale it adaptively from an exponential moving average of the observed nonzero-signal fraction after a measurement-only probe period. On two development targets used for hyperparameter selection, a fixed correct-pair bonus improved peak TM-score on the sparse-signal target 2GCS but substantially degraded it on the already-responsive target 2GDI. Adaptive scaling moderated this trade-off in the development runs, retaining much of the fixed-bonus improvement on 2GCS while reducing its degradation on 2GDI; these pilot results motivate, rather than replace, held-out comparative evaluation. Separately, we identify and correct an error in how multi-chain benchmark structures were filtered for the secondary-structure reward. Replacing a sequence-midpoint approximation with residue-level chain assignments from the source structures showed that one target had been incorrectly treated as multi-chain and that another contained only inter-chain pairs under the old retained set. Targets affected by this correction were rerun using the corrected pair definitions, and only the corrected runs are retained in the final benchmark results. These results emphasize that auxiliary structural rewards depend critically on accurate target construction.

1 Introduction

RNA inverse design is the problem of identifying nucleotide sequences that satisfy a desired structural constraint. Historically, much of computational RNA design has focused on secondary structure, where a target pairing pattern is specified and candidate sequences are optimized so that a folding model recovers that pattern. Reinforcement learning has also been applied in this setting: LEARNA, for example, learns a policy that sequentially constructs sequences for specified secondary structures **?**.

More recent work has shifted toward direct conditioning on RNA tertiary structure. gRNAde uses geometric deep learning to generate sequences from one or more 3D RNA backbones Joshi et al. [2024], while RiboDiffusion applies a generative diffusion model to tertiary-structure-based inverse folding **?**. These approaches reflect an important distinction between recovering a native sequence and designing a sequence that reproduces a target fold: multiple sequences may be structurally compatible with the same backbone, so sequence recovery is an imperfect surrogate for structural fidelity.

RIDER Hu et al. [2026] addresses this distinction by fine-tuning a diffusion-based RNA sequence generator with reinforcement learning against structure-based self-consistency rewards. In this setting, re-

ward formulation becomes part of the model-design problem: the policy can only optimize structural properties that are represented in the reward signal.

RIDER's original RL stage optimizes exclusively for tertiary structural similarity:

$$R_{\text{RIDER}} = -\left(\tfrac{1}{2}\,\text{RMSD}\right)^2 + (5\,\text{GDT\_TS})^2 + \text{bonus}. \tag{1}$$

RMSD (root-mean-square deviation, in Å) measures average atomic displacement between the predicted and target C4′ coordinates after optimal superposition. GDT_TS (global distance test total score) is the fraction of residues superimposing within a set of distance thresholds (1, 2, 4, 8 Å), GDT_TS $\in [0, 1]$. The coefficients $\frac{1}{2}$ and 5 are hand-tuned normalization constants bringing the two terms, which live on different numeric scales, into comparable magnitude prior to squaring; they are not derived quantities. bonus is an additional linear term that activates once GDT_TS > 0.45 or RMSD < 3 Å, increasing reward separation among candidates after a high-similarity threshold is reached.

RhoFold, the folding oracle used by RIDER Shen et al. [2024], additionally computes a secondary-structure base-pairing prediction via its `ss_head` module on every forward pass. This output is never consumed by RIDER's reward function; it is computed and immediately discarded on every one of the thousands of oracle calls made during a single RL run. Because RNA tertiary structure is organized around secondary-structure elements, formation of the intended 3D fold may benefit from an explicit signal about the underlying base-pairing, information RIDER's data pipeline already extracts and stores (in dot-bracket notation, in the `sec_struct_list` field) but does not use during RL fine-tuning.

1.1 Study Objective and Hypotheses

This study evaluates whether secondary-structure reward shaping can improve RIDER's optimization behaviour without sacrificing tertiary structural fidelity. We test two related hypotheses.

H1: Complementary structural supervision. Explicitly rewarding recovery of the target base-pair pattern will increase secondary-structure agreement and may improve tertiary structural fidelity on targets for which the original RIDER reward provides weak or incomplete local structural guidance.

H2: Adaptive shaping for sparse secondary-structure signal. When exact base-pair recovery is rare, a fixed F1 term provides little differentiation among sampled sequences. Amplifying rare correct-pair events should make partial successes more useful to PPO, while adaptively tapering that amplification as correct-pair recovery becomes common should reduce the risk of over-weighting secondary structure on already well-performing targets.

2 Secondary-Structure Reward Term

For each generated sequence, RhoFold's `ss_head` module outputs raw, unactivated logits $L \in \mathbb{R}^{n\times n}$, one value per residue pair. The implementation first applies sigmoid elementwise and then symmetrizes the resulting probabilities,

$$P = \tfrac{1}{2}\left(\sigma(L) + \sigma(L)^\top\right), \tag{2}$$

with the diagonal set to zero because a residue cannot pair with itself. This order matches the implemented decoder: sigmoid is applied before symmetrization.

Predicted pairs are then obtained using a Nussinov-style dynamic program **?**. Candidate pairs must satisfy $P_{ij} \geq 0.5$ and $j - i > 3$, and the dynamic program selects a non-crossing, one-partner-per-residue set maximizing total retained pairing probability. Thus, the loop constraint requires at least three enclosed residues between paired positions. No constraint is placed on nucleotide identity: RhoFold's `ss_head` is not queried for base-type compatibility, and the decoder does not separately filter for canonical Watson–Crick or wobble pairs.

The decoder settings are fixed and distinct from the reward-shaping hyperparameters introduced below. The 0.5 threshold is the natural binary decision boundary after sigmoid activation and was not tuned on the benchmark. The three-residue minimum hairpin-loop constraint follows established RNA secondary-structure decoding practice. Symmetrization enforces the undirected nature of a base-pair relation, and the Nussinov-style scoring formulation provides a deterministic non-crossing decode. These settings were held fixed throughout all experiments and are separate from the six reward-shaping hyperparameters (λ_{SS}, s_{base}, f_{target}, β, p, and c)

selected using the development-target protocol in Section 4.3.

Target pairs are derived from `sec_struct_list[0]`, a dot-bracket string computed once, prior to training, from the deposited crystal structure via X3DNA **?** (invoked through the dataset's `pdb_to_tensor` utility with `return_sec_struct=True`); this is the same field parsed into the (i, j) index-pair set used throughout this paper. If a target has zero valid pairs after parsing, R_{SS} is defined as 0 by convention (Section 3 treats this as a genuine, not merely edge-case, scenario). The predicted pair set is compared against the target's known base pairs using the base-pair F1 score:

$$R_{\text{SS}} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \tag{3}$$

where TP, FP, and FN denote true-positive, false-positive, and false-negative predicted base pairs relative to the target.

This term is combined with the existing reward as:

$$R_{\text{combined}} = R_{\text{RIDER}} + \lambda_{\text{SS}} \cdot R_{\text{SS}} \tag{4}$$

where λ_{SS} controls the contribution of secondary-structure recovery relative to the existing tertiary reward.

2.1 Target Scope

Two categories of Das et al. benchmark [Das et al., 2010] structures require special handling.

Pseudoknotted targets (1F27, 1L2X): the Nussinov-style decoder used to obtain predicted pairs guarantees a non-crossing structure by construction, an inherent property of the dynamic programming formulation, not an implementation gap. Pseudoknotted targets fall outside the support of the present reward formulation.

Multi-chain-flagged targets: An initial implementation approximated chain membership using a sequence-midpoint split, which assumes two equal-length chains and does not account for preprocessing-induced index changes. We replaced this approximation with a chain-aware filter. Per-residue chain identity was extracted from the source crystal structure using MDAnalysis **?**; atom-level annotations were collapsed to a per-residue assignment, with residues containing disagreeing atom-level chain IDs flagged rather than silently assigned. The raw residue sequence was then globally aligned to the corresponding preprocessed dataset representation to establish residue-level correspondence after any preprocessing-induced removals or modifications. A target pair (i, j) was retained for the present secondary-structure reward only when both mapped residues belonged to the same chain. This defines an explicitly intra-chain target representation; inter-chain base pairs are outside the support of the current reward. The resulting target reclassification is reported in Section 5.1.

3 Sparse Secondary-Structure Reward Regime

Initial experiments applying Equation 4 directly revealed that R_{SS} remains at exactly 0 for the entire training run on several targets, independent of λ_{SS} magnitude (tested up to $\lambda_{\text{SS}} = 5.0$, an order of magnitude above the effective range, with no change in outcome). Because R_{SS} is a ratio, it registers a legitimate hard zero when predicted and target pair sets share no overlap, unlike RMSD or GDT_TS, which are continuous by construction. When every sampled sequence in a training batch scores $R_{\text{SS}} = 0$, the term provides no differentiation among trajectories, and scaling λ_{SS} cannot create information that is not present in the batch.

4 Correct-Pair Shaping Bonus

To address this sparse-signal regime, we introduce a secondary shaping term that rewards individual correct target pairs independently of overall precision or recall:

$$R_{\text{SS-bonus}} = \begin{cases} \min(TP, c) \times s_{\text{bonus}}, & \text{if } TP > 0 \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where s_{bonus} is a scaling coefficient and c is a cap on the number of pairs counted per sample (Section 4.2). Note that this term is identically zero whenever $TP = 0$ for every sample in a batch, exactly as R_{SS} is; the shaping term cannot create information when every candidate has $TP = 0$. Instead, it increases reward separation once rare $TP > 0$ events appear, rather than relying solely on a reward component that provides little or no differentiation among sampled sequences in the sparse-signal regime.

The correct-pair bonus is used as an auxiliary shaping term rather than as a replacement for R_{SS}. A reward based only on true-positive count would encourage recovery of target pairs without penalizing additional false-positive pairings. The F1 term retains sensitivity to both precision and recall, while the bonus selectively amplifies rare correct-pair events once they appear.

4.1 Adaptive Bonus Scaling

A fixed s_{bonus} was found to improve outcomes on targets where R_{SS} signal was largely absent under Equation 4 alone, but to degrade tertiary accuracy on targets where real signal was already present, plausibly by over-weighting secondary-structure optimization once it was no longer needed (Section 5). We therefore scale s_{bonus} adaptively, based on the observed fraction of sampled sequences per epoch with at least one correct pair:

$$s_{\text{bonus}} = s_{\text{base}} \times \max\left(0,\ \min\left(1,\ 1 - \frac{\bar{f}_{\text{nz}}}{f_{\text{target}}}\right)\right) \tag{6}$$

where s_{base} is the maximum (ceiling) bonus scale, f_{target} is a target nonzero-fraction threshold, and $\bar{f}_{\text{nz}}$ is an exponential moving average of the per-epoch nonzero fraction $f_{\text{nz}}^{(e)}$:

$$\bar{f}_{\text{nz}}^{(e)} = \begin{cases} f_{\text{nz}}^{(e)}, & e = 0 \\ \beta\, \bar{f}_{\text{nz}}^{(e-1)} + (1-\beta)\, f_{\text{nz}}^{(e)}, & e > 0 \end{cases} \tag{7}$$

4.1.1 Measurement-Only Probe Period

An initial version seeded $\bar{f}_{\text{nz}}$ from a per-target lookup table of previously-measured baselines. This does not generalize to unseen targets and was replaced with a measurement-only probe: for the first p epochs of every run, s_{bonus} is forced to 0 while R_{SS} and TP continue to be measured normally via the $\lambda_{\text{SS}} \cdot R_{\text{SS}}$ pathway, allowing $\bar{f}_{\text{nz}}$ to be established from genuine per-target observation before the bonus term ever contributes to the reward.

4.2 Bonus Pair Cap

Because $R_{\text{SS-bonus}}$ is unbounded in TP, a single sample with an unusually large number of correct pairs can contribute a reward far outside the range produced by R_{RIDER}'s own bonus term (bounded to approximately $[0, 60]$). Such an outlier can, under the epoch-level reward baseline used for PPO's **?** advantage estimate, disproportionately shift the baseline for subsequent epochs. We cap the number of pairs counted per sample at c, chosen such that $c \times s_{\text{base}}$ remains within the same order of magnitude as R_{RIDER}'s existing bonus range.

Parameter	Value / Role
λ_{SS}	0.2; weight on the F1 term
s_{base}	8.0; ceiling bonus scale
f_{target}	0.15; nonzero-fraction taper threshold
β	0.8; EMA smoothing factor
p	5; probe period length, epochs
c	7; correct-pair cap per sample

Table 1: Hyperparameters introduced by the secondary-structure reward extension.

4.3 Hyperparameter Selection Protocol

Because six hyperparameters govern this reward extension, tuning all of them while observing performance across the full benchmark and reporting that same benchmark as the final result would constitute tuning to the test set. Hyperparameter values (Table 1) were instead selected exclusively through iterative development on two *development targets*, 2GCS and 2GDI, chosen to represent opposite ends of the baseline signal-density spectrum: 2GCS was near-dead under the unmodified combined reward (nonzero-fraction $\approx 1\%$), while 2GDI already exhibited strong baseline signal ($\approx 90\%$). All six hyperparameters were frozen at the values shown before evaluating on any remaining Das14 target; no hyperparameter was subsequently adjusted based on performance elsewhere. Because 2GCS and 2GDI directly informed hyperparameter selection, they are reported separately throughout this paper (Table 3) and are not included among held-out generalization statistics.

5 Results

5.1 Chain-Aware Target Validation

The chain-aware validation produced three unchanged and three changed target-pair sets. 1CSL is a genuine two-chain structure with one pair reclassified. 1Q9A was in fact single-chain: the midpoint heuristic had misread long-range intra-chain pairs as

Target	Chains	Mid.	True	Sym. diff	Action
1XPE	2	0	0	0	No rerun
1LNT	2	0	0	0	No rerun
354D	2	0	0	0	No rerun
1CSL	2	1	0	1	Rerun
1Q9A	1	0	10	10	Rerun
1X9C	4	18	0	18	Rerun

Table 2: Reclassification of benchmark structures originally flagged as multi-chain. "Mid." and "True" denote target pairs retained under the midpoint approximation and corrected chain-aware filter, respectively.

Target	Condition	Peak TM
2GCS	Baseline	0.672
	+F1	0.643
	+Fixed bonus	0.744
	+Adaptive	0.716
2GDI	Baseline	0.614
	+F1	0.585
	+Fixed bonus	0.484
	+Adaptive	0.566

Table 3: Peak TM-score across reward conditions on the two development targets.

Target	Peak TM	Note
4FE5	0.802	
1ET4	0.427	
2R8S	0.748	
1XPE	0.715	
1LNT	0.022	
2OEU	0.222	
354D	0.002	
2GCS*	0.716	development
2GDI*	0.566	development
1CSL†	0.198	corrected
1Q9A†	0.531	corrected
1X9C†	0.236	corrected

Table 4: Final retained peak TM-score for each applicable target under the adaptive configuration. *Development targets used for hyperparameter selection; †targets rerun after chain-aware pair-set correction. Earlier superseded or incomplete runs are not treated as independent replicates.

inter-chain and discarded all ten real target pairs, so correction restores a valid signal. 1X9C is a genuine four-chain structure; all eighteen pairs the midpoint method had retained are inter-chain, so correction removes a fully false signal, leaving zero intra-chain target pairs. Because the corrected pair definitions supersede the earlier midpoint-based targets, 1CSL, 1Q9A, and 1X9C were rerun and only the corrected runs are retained in the final benchmark results.

5.2 Reward-Shaping Comparison on Development Targets

Table 3 compares four reward conditions on the two development targets, tertiary-only baseline ($\lambda_{SS} = 0$), the F1 term alone ($\lambda_{SS} = 0.2$, no bonus), a fixed non-adaptive bonus ($s_{bonus} = 8.0$ throughout training, no taper), and the final adaptive configuration. These are the only two targets for which all four reward conditions were evaluated during method development; extending a controlled condition-wise comparison across the held-out benchmark remains future work (Section 7).

Peak TM-score is reported here for direct comparability with earlier development-phase logging. Given the training instability documented in Section 5.5, peak TM alone should not be treated as conclusive evidence of a stable improvement.

5.3 Full Benchmark Results

Table 4 reports one final retained peak TM-score for each applicable Das14 target under the final adaptive configuration. 1F27 and 1L2X are excluded per Section 2.1. When a target was rerun because of a corrected target definition or an incomplete technical attempt, the earlier run was superseded rather than treated as an independent replicate. 2GCS and 2GDI are development targets (Section 4.3) and are excluded from held-out generalization claims.

Table 4 is descriptive: it establishes the range of outcomes obtained from the final retained runs under the adaptive configuration, but it does not by itself estimate an adaptive-versus-baseline treatment effect because the full four-condition com-

parison has not been completed on the held-out targets. Accordingly, no benchmark-wide claim of rescue or improvement is made from these scores alone. The development-target comparison in Table 3 shows that reward-shaping effects are strongly target-dependent, while controlled held-out condition comparisons are required to determine whether adaptive scaling generalizes.

5.4 Chain-Boundary Correction

The chain-boundary correction is treated as a target-definition fix rather than as a replicated experimental ablation. The original midpoint approximation was identified during validation as an inaccurate representation of chain membership for 1CSL, 1Q9A, and 1X9C (Table 2). Those targets were rerun using the corrected residue-level chain assignments, and the corrected runs replace the earlier midpoint-based runs in Table 4.

This distinction is particularly important for 1X9C. Under the midpoint approximation, eighteen retained target pairs were inter-chain even though the implemented reward is defined over an intra-chain target representation. Under the corrected configuration, the intra-chain target pair set is empty, so the secondary-structure terms are identically zero for this target. The correction therefore removes a mis-specified auxiliary objective; however, because the earlier run is treated as superseded development output rather than an independent replicate, we do not use the before/after TM-score difference as a formal estimate of the correction's effect.

5.5 Training Instability

Training instability was assessed using a descriptive, post hoc diagnostic intended to distinguish sustained deterioration from isolated per-sample fluctuations. For each run, the first 10% of logged steps were excluded as an early-training warmup period. A 15-point rolling mean was then computed over the remaining TM-score trajectory, and a run was flagged for instability when the difference between its overall peak TM-score and the minimum of the smoothed post-warmup trajectory exceeded 0.20.

These diagnostic settings were not specified a priority. An initial criterion based on the unsmoothed minimum and a threshold of 0.15 produced false-positive flags on trajectories judged stable by visual inspection, including 1XPE and stable 4FE5 runs. The smoothing window, warmup exclusion, and 0.20 threshold were therefore selected iteratively to reduce sensitivity to isolated low-scoring samples. Accordingly, the criterion is used only as a descriptive flag for runs warranting closer inspection and is not interpreted as a pre-specified statistical test of training instability.

6 Discussion

The development-target comparison supports the motivation for adaptive shaping more strongly than it supports a claim of universal performance improvement. On 2GCS, where baseline correct-pair events were rare, direct F1 weighting alone did not improve peak TM-score, whereas a fixed correct-pair bonus increased it from 0.672 to 0.744. On 2GDI, where secondary-structure signal was already common, the same fixed bonus reduced peak TM-score from 0.614 to 0.484. Adaptive scaling produced intermediate outcomes in the development runs (0.716 and 0.566, respectively), consistent with the intended role of tapering. Because both targets were used during hyperparameter development and the four-condition comparison was not designed as a replicated seed study, these results are mechanistic pilot evidence rather than a held-out efficacy estimate.

The chain-boundary analysis provides a separate methodological result. The secondary-structure objective is only meaningful when its target pair set corresponds to the representation being optimized. For 1X9C, the midpoint approximation exposed the policy to an auxiliary objective composed entirely of inter-chain pairs even though the implemented reward evaluates an intra-chain target representation. For 1Q9A, the same approximation incorrectly removed ten valid intra-chain pairs. These errors were corrected before the final reported runs, and the superseded midpoint-based runs are not treated as replicated evidence of a treatment effect. The main conclusion from this analysis is therefore methodological: target construction can determine whether an auxiliary reward is informative, misleading, or effectively absent.

A remaining methodological question is whether hard decoding is the best interface to RhoFold's secondary-structure head. The present formulation converts continuous pair probabilities into a discrete

pair set before computing F1 and the correct-pair bonus. This provides an interpretable structural objective but can create a sparse regime when no decoded pair overlaps the target. A future alternative is a continuous overlap or soft-F1 objective computed directly from pair probabilities, which could provide denser supervision without requiring rare threshold-crossing events.

7 Limitations

- **Baseline comparison scope.** The complete four-condition reward-shaping comparison (Table 3) is currently limited to the two development targets. These targets informed hyperparameter selection and therefore cannot support held-out generalization claims. Extending controlled condition-wise comparisons across the held-out benchmark remains necessary.
- **Incomplete secondary-structure outcome logging.** SS-F1 and nonzero-fraction were not retained in the summary tables for the early development runs. Until these quantities are re-extracted from the full histories, H1's predicted increase in secondary-structure agreement cannot be evaluated directly from Table 3.
- **Hard-decoded reward interface.** The current reward thresholds RhoFold pair probabilities and decodes a non-crossing pair set before computing F1. A continuous probability-based overlap objective could provide denser supervision and has not yet been compared against the hard-decoded formulation.
- **Target-representation scope.** The current Nussinov-style decoder does not represent pseudoknots, excluding 1F27 and 1L2X, and the chain-aware filter defines an intra-chain secondary-structure target, so inter-chain base pairs are not rewarded.
- **Post hoc instability diagnostic.** The instability criterion was developed after inspection of training trajectories and calibrated to reduce false-positive flags on visually stable runs. It is descriptive rather than inferential, and the frequency of flagged runs should not be interpreted as a formally estimated collapse probability.
- **Single-run final reporting.** The benchmark table reports one final retained completed run per target. Earlier reruns and superseded target definitions are not treated as independent replicates, so run-to-run variability under the finalized configuration is not quantified.

8 Conclusion

RhoFold computes a secondary-structure prediction on every forward pass within RIDER's RL loop, yet the original tertiary reward does not use this signal. We augment RIDER with a base-pair F1 term and a capped correct-pair shaping bonus whose scale is estimated per target after a measurement-only probe period and tapered as correct-pair events become common. Development experiments reveal the central trade-off motivating this design: a fixed bonus improved peak TM-score on sparse-signal 2GCS but degraded it on already-responsive 2GDI, while adaptive scaling moderated the trade-off in the development runs. These two development targets were used for hyperparameter selection, so the result motivates held-out comparative testing rather than establishing benchmark-wide superiority. Secondary-structure metrics from the existing training histories are also required to test directly whether the added objective improves base-pair agreement. Separately, replacing an approximate midpoint chain filter with residue-level chain assignments corrected mis-specified secondary-structure targets; affected targets were rerun and only the corrected runs are retained in the final benchmark. The current evidence therefore supports two narrower conclusions: adaptive shaping is a plausible mechanism for managing target-dependent reward sparsity, and accurate target construction is essential when introducing auxiliary structural objectives into RNA inverse-design RL.

References

Chaitanya K. Joshi, Arian R. Jamasb, Ramon Viñas, Charles Harris, Simon Mathis, Alex Morehead, Rishabh Anand, and Pietro Liò. gRNAde: Geometric deep learning for 3D RNA inverse design. In *Proceedings of the 41st International Conference on Machine Learning (ICML 2024)*, Vienna, Austria, 2024.

Zhihang Hu et al. RIDER: Reinforcement learning

for RNA inverse design with equivariant representations. In *Proceedings of the 14th International Conference on Learning Representations (ICLR 2026)*, Singapore, 2026.

Tao Shen, Zhihang Hu, Siqi Sun, Di Liu, Felix Wong, Jiuming Wang, Jiayang Chen, Yixuan Wang, Liang Hong, Jin Xiao, et al. Accurate RNA 3D structure prediction using a language model-based deep learning approach. *Nature Methods*, 21:2287–2298, 2024. doi: 10.1038/s41592-024-02487-0.

Rhiju Das, John Karanicolas, and David Baker. Atomic accuracy in predicting and designing noncanonical RNA structure. *Nature Methods*, 7: 291–294, 2010. doi: 10.1038/nmeth.1433.