DO PLAUSIBLE STRUCTURES GUARANTEE COHERENT DYNAMICS? CONTEXTUAL ANOMALY LOCALIZATION IN PROTEIN TRAJECTORIES

Anonymous authors
Paper under double-blind review

ABSTRACT

Understanding how protein conformations evolve over time is essential for revealing the molecular mechanisms underlying biological function. Yet a trajectory can pass framewise structural checks and match a reference conformational distribution while containing temporally misplaced transitions, potentially misleading interpretations of conformational transition pathways. The challenge is to distinguish such contextual inconsistencies from legitimate fluctuations and conformational changes, which cannot be resolved by the magnitude of structural change alone. We introduce ProTAL, an offline framework that localizes trajectory anomalies by learning the compatibility between conformational transitions and their temporal context. ProTAL combines bidirectional latent predictions with residue-pair geometry across multiple temporal scales, integrating expectations of molecular evolution with direct evidence of structural change. Learning from unperturbed trajectories and controlled temporal corruptions, it localizes suspicious transitions without requiring paired reference trajectories at inference. On our ATLAS-derived protein-trajectory benchmark, ProTAL achieves substantially higher anomaly detection success rates than the conventional baselines evaluated. ProTAL adds transition-level diagnostics to structural and ensemble evaluation, helping researchers identify potentially misleading trajectory segments before drawing mechanistic conclusions. The code is available at the link: https://anonymous.4open.science/r/ProTAL-6EED.

1 INTRODUCTION

Protein function depends on how conformations evolve over time Boehr et al. (2006); Henzler-Wildman et al. (2007). Molecular dynamics (MD) simulations reveal these motions, but their computational cost has motivated learning-based alternatives. These include conformational sampling methods such as Str2Str, AlphaFlow, and structure language modelsLu et al. (2024); Jing et al. (2024a); Lu et al. (2025), as well as trajectory generators such as MDGen Jing et al. (2024b). Reliable trajectory evaluation is therefore essential, as mistimed conformational transitions can distort inferred molecular events and mislead mechanistic interpretations.

Existing evaluations primarily assess the structural plausibility of individual frames Chen et al. (2010), agreement with reference conformational ensembles Jing et al. (2024a), or temporal statistics Prinz et al. (2011); Jing et al. (2024b). Structural checks can identify implausible conformations but cannot determine whether a plausible one fits between its neighboring frames. Likewise, shuffling the same frames leaves their empirical conformational distribution unchanged. Autocorrelations and state-transition statistics characterize temporal behavior, but these aggregate measures do not directly pinpoint anomalous transitions. These limitations motivate a complementary diagnostic that uses the trajectory's own context to localize suspicious transitions without a paired reference trajectory at inference.

However, global change magnitude alone does not determine temporal compatibility. Large-scale motions can be functional Henzler-Wildman et al. (2007), while small changes may still be mistimed. Residue-pair geometry can reveal relative motion patterns hidden by similar global change magnitudes. We therefore introduce **ProTAL** (Protein Trajectory Anomaly Localizer), which com-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

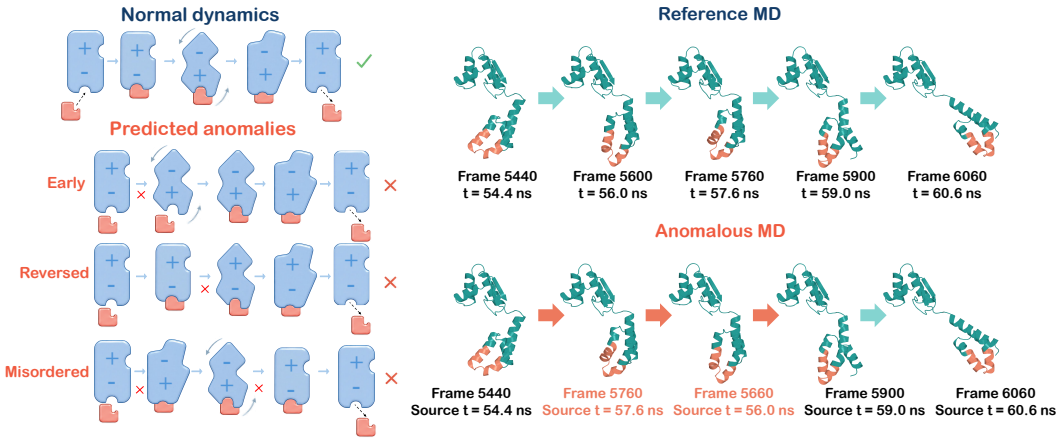


Figure 1: **Temporal context matters.** A reference sequence is contrasted with mismatches in timing (Early), state-marker arrangement (Reversed), and intermediate-state ordering (Misordered). Crosses indicate contextual mismatches. The coral partner and $+/-$ markers are schematic context, not ProTAL inputs; the examples are neither model predictions nor universal physical-validity rules.

bines contextual predictions with residue-level geometry. Its Motion Predictor assesses contextual compatibility through bidirectional latent-space prediction across multiple horizons, while its Transitions Reader tracks changes in residue-pair geometry and aggregates them over time. Fusing these branches enables offline localization of suspicious transitions without a paired reference trajectory.

We evaluate ProTAL on an ATLAS-derived benchmark Vander Meersche et al. (2024), with explicit localization and false-alarm criteria. On identical held-out windows, ProTAL achieves 55.00% macro-averaged recall within a three-edge neighborhood, compared with 12.62% for the strongest evaluated conventional baseline, Transition innovation (TI). Conditional MD further supports selected detections missed by conventional diagnostics. Our main contributions are as follows:

1. We introduce ProTAL to address the gap in local temporal diagnostics left by framewise structural checks and aggregate statistical evaluation, localizing suspicious transitions without paired reference trajectories at inference.
2. We combine multi-horizon bidirectional latent-space prediction with temporal modeling of residue-pair geometry to identify anomalous transitions that global change magnitudes alone cannot reliably distinguish.
3. Through comparative and ablation experiments on the ATLAS-derived benchmark and conditional MD validation, we demonstrate improved localization and complementary detection of transitions missed by conventional diagnostics.

2 Related Work

Protein conformational ensembles. Learning-based samplers amortize the cost of exploring protein conformations. Boltzmann generators learn transformations for equilibrium sampling Noé et al. (2019); Str2Str uses score-based perturbation and denoising around an input structure Lu et al. (2024); and AlphaFlow adapts structure predictors through flow matching Jing et al. (2024a). Structure language models instead generate discrete structural tokens, including the masked-diffusion ESMDiff formulation Lu et al. (2025). BioEmu further combines structural, simulation, and experimental stability data to model equilibrium ensembles Lewis et al. (2025). These advances improve access to conformational diversity and populations. Their sampling order, however, need not represent elapsed physical time: agreement with an equilibrium distribution alone does not determine transition kinetics. ProTAL addresses temporal compatibility when observations are interpreted as an ordered trajectory.

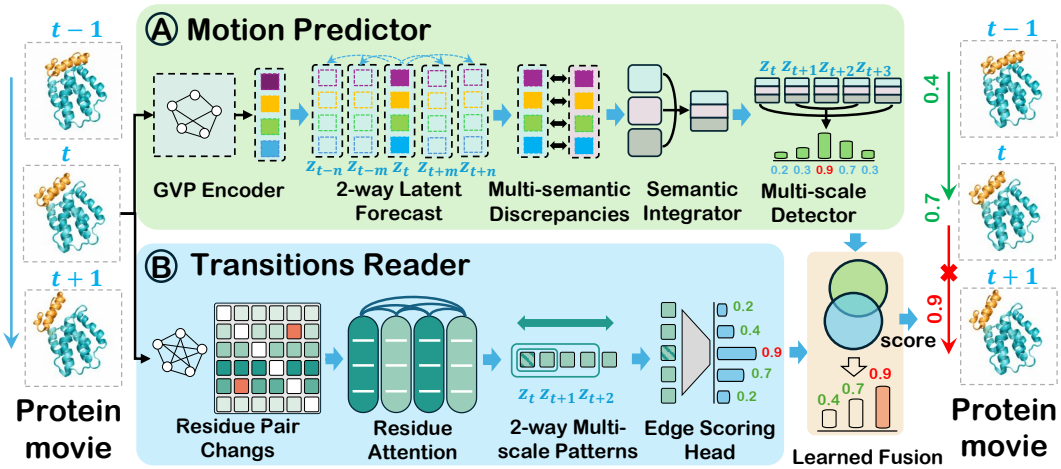


Figure 2: **ProTAL overview.** **(A)** The Motion Predictor aligns bidirectional, multi-horizon prediction errors to temporal edges, integrates latent and geometric changes, and aggregates evidence across scales. **(B)** The Transitions Reader aggregates residue-pair changes into residue tokens and models their temporal patterns. Learned Fusion combines both branch scores. Structures and scores are schematic; z_t represents a frame and q_t a transition.

Dynamical models and kinetic assessment. Learning temporal evolution requires additional information. Timewarp learns time-coarsened proposals for equilibrium MCMC sampling Klein et al. (2023), whereas implicit transfer operators learn conditional dynamics at multiple time resolutions Schreiner et al. (2023). MDGen models complete trajectories and supports simulation, interpolation, and upsampling Jing et al. (2024b). These methods already assess aspects of temporal behavior; MDGen, for example, evaluates torsional relaxation and Markov-state fluxes on tetrapeptides. More generally, Markov-state models and VAMPnets estimate kinetic structure from time-lagged observations Prinz et al. (2011); Mardt et al. (2018). Such analyses characterize dynamical statistics, while ProTAL produces a score for each observed transition. This local diagnostic complements ensemble and kinetic evaluation, without replacing either or estimating a full trajectory likelihood.

Predictive representations. Time-lagged autoencoders learn slow molecular coordinates (Wehmeyer & Noé, 2018), while joint-embedding architectures learn representations by predicting unobserved image or video content (Assran et al., 2023; Bardes et al., 2024). LeWorldModel studies latent dynamics and physical surprise in visual environments (Maes et al., 2026). ProTAL combines predictive discrepancies with residue-pair geometry for supervised transition localization.

Temporal anomaly detection. Anomaly Transformer uses association discrepancy (Xu et al., 2022); DCdetector learns contrasting patch representations Yang et al. (2023); and DADA uses adaptive bottlenecks and adversarial decoders for cross-domain detection Shentu et al. (2025). Protein trajectories additionally require geometric features and residue masking while accommodating legitimate stochastic motion. Because point adjustment can inflate detection scores Kim et al. (2022), we report exact-edge W1 and bounded-tolerance W3 recall, clean FPR, and alarm FDP. Section 4.3 evaluates adapted learned methods; Appendix D provides further discussion.

3 Method

Let $\mathcal{X} = \{(X_t, \tau_t)\}_{t=0}^{T-1}$ be a trajectory window, with Cα coordinates $X_t \in \mathbb{R}^{N\times 3}$, observation times τ_t, residue identities, original sequence positions, and a validity mask. ProTAL returns one anomaly score for each edge $X_t \to X_{t+1}$. As an offline detector, it uses observations on both sides of an edge. Figure 2 shows its two pathways: contextual latent prediction and direct residue-pair analysis. The following describes their essential operations; complete features and network specifications are in Appendix A.

3.1 CONTEXTUAL MOTION PREDICTION

A GVP-style graph encoder Jing et al. (2021) maps each conformation to a latent state $z_t \in \mathbb{R}^d$. Geometry, residue identity, and sequence position inform scalar–vector message passing, followed by graph pooling and projection. The resulting state is invariant to proper rigid rotations and translations while retaining chirality-sensitive information. Two independently parameterized Transformers Vaswani et al. (2017) then read the latent sequence causally in forward and reversed order, conditioned on directional increments, chirality summaries, and timestamps. For horizons $\mathcal{H} = \{1, 2, 4, 8\}$, they directly predict latent displacements:

$$\begin{aligned} \widehat{z}^{+}_{i+h|i} &= \mathrm{sg}(z_i) + P^{+}_h(c^{+}_i), \\ \widehat{z}^{-}_{i|i+h} &= \mathrm{sg}(z_{i+h}) + P^{-}_h(c^{-}_{i+h}), \end{aligned} \tag{1}$$

where c^+ and c^- are directional context representations and sg stops gradients. Predictions are direct at each horizon, and targets are also detached. This supplies contextual evidence without reconstructing Cartesian trajectories.

To convert forecasts into transition evidence, let $\epsilon^{\sigma}_{i,h}$ be the mean squared latent prediction error for direction σ. A horizon-h forecast crosses h edges, so we distribute its error over those edges and average overlapping forecasts:

$$E^{\sigma}_{t,h} = \frac{1}{|\mathcal{I}_{t,h}|} \sum_{i \in \mathcal{I}_{t,h}} \epsilon^{\sigma}_{i,h}, \qquad e^{\sigma}_{t,h} = \log\!\left(1 + E^{\sigma}_{t,h} / \nu_{\sigma,h}\right), \tag{2}$$

where $\mathcal{I}_{t,h} = \{i : 0 \le i < T - h,\ i \le t < i + h\}$ and $\nu_{\sigma,h} > 0$ is estimated from clean prediction errors. The eight directional and horizon channels form e_t.

Prediction error alone cannot distinguish unusual motion from a contextual inconsistency. The Semantic Integrator therefore combines e_t, the observed latent increment Δz_t, and local geometric changes g_t describing pair distances and chirality. For each width $w \in \{15, 7, 3, 1\}$, their projections are added and summarized by centered mean/max pooling. The Multi-scale Detector reads the resulting contexts jointly to produce s^A_t. This preserves both discrepancy magnitude and its temporal pattern; the exact integration and scale-specific readouts are given in Appendix A.2.

3.2 RESIDUE GEOMETRY AND SCORE FUSION

Global latents can obscure where structural changes occur. The Transitions Reader therefore starts from distances $d_{t,ij} = \|x_{t,i} - x_{t,j}\|_2$ and increments $\delta_{t,ij} = d_{t+1,ij} - d_{t,ij}$. For each valid unordered pair, a fixed descriptor combines distance and soft-contact changes, deviations from preceding and following increments, and positional information. Pairwise distances avoid dependence on rigid alignment. These quantities describe changes per sampled interval; feature definitions are in Appendix A.3.

A shared MLP embeds each pair, and mean/max aggregation over incident pairs forms a token for each residue. Residue attention models interactions before pooling the tokens into a transition representation $q^{(0)}_t$. Four residual temporal convolution blocks with dilations $(1, 2, 4, 8)$ then capture earlier and later transition patterns, and an edge head produces s^B_t. All residue aggregations respect the validity mask. Retaining residue tokens until after attention complements the Motion Predictor's pooled frame representation.

The final anomaly score is a learned affine combination:

$$s_t = \beta_A s^A_t + \beta_B s^B_t + b. \tag{3}$$

The weights are unconstrained, and s_t is a score rather than a calibrated probability.

3.3 TRAINING AND INFERENCE

Each corrupted training window is paired with its unmodified counterpart. Clean windows train latent dynamics; frame swaps, block swaps, distant-frame replacements, replica splices, and matched

splices provide explicit anomaly labels. The complete detector is therefore supervised. Its objective is

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_{\text{clean}} + \text{sg}(\eta)\mathcal{L}_{\text{aux}} + \lambda_{\text{det}}\mathcal{L}_{\text{det}}, \\ \mathcal{L}_{\text{det}} &= \mathcal{L}_{\text{hier}} + \mathcal{L}^{\text{bal}}_{\text{BCE}}(s) + 0.5\mathcal{L}^{\text{bal}}_{\text{BCE}}(s^B). \end{aligned} \tag{4}$$

Here $\mathcal{L}_{\text{clean}}$ combines mean and smooth worst-head prediction errors. The budgeted auxiliaries preserve representation diversity, distinguish correct-time targets, and anchor states and increments to geometry. Detection combines family-balanced classification with multiscale objectives emphasizing localization at low false-alarm budgets and paired clean/corrupted score constraints. Direct BCE supervises original positive edges, while the final localization objective allows a centered three-edge neighborhood. Appendix B specifies the losses and gradient controls.

Training first fits clean dynamics and then enables supervised detection. Gradient controls are detailed in Appendix B. At inference, only the candidate window and its metadata are required. Binary alarms use $s_t \geq \theta$, with θ fixed at the higher empirical 99th percentile of clean scores from separate calibration proteins. This sets an operating point; held-out FPR is measured rather than assumed to be 1%.

4 Experiments

We assess localization of controlled corruptions, false-alarm behavior, and whether MD supports detections missed by conventional diagnostics.

4.1 Experimental Setup

Data and model. We train on 200 ATLAS proteins comprising 12,780 windows Vander Meersche et al. (2024). A separate 39-protein subset provides 20 proteins for calibration and checkpoint selection (1,098 windows) and 19 for conventional-diagnostic comparisons (1,278 windows). Test82 supports learned-detector comparisons, ablations, and the time-warp challenge. Each window contains 32 frames at 1 ns intervals and up to 64 Cα atoms, yielding 31 transition scores. Five corruption families alter temporal order or continuity using MD-sampled conformations; matched splicing additionally constrains boundary compatibility. Labels indicate constructed temporal perturbations. The checkpoint is selected by calibration W3 recall and frozen for evaluation. Appendix C provides the protocols.

Comparators and evaluation. We compare four static controls, four local dynamic controls, and reference-assisted Transition innovation (Table 1). W3 recall detects a labeled edge if an alarm falls on it or either neighbor; W1 requires the exact edge. Consecutive labels remain separate units, and macro recall equally weights the five families. Unless stated otherwise, thresholds are fixed by clean calibration q99. The conventional-diagnostic audits use 30 common finite-score positions per window. We distinguish clean-data FPR, $\text{FP}/(\text{FP} + \text{TN})$, from the false-discovery proportion among alarms, $\text{FDP} = \text{FP}/(\text{TP} + \text{FP})$. Discrete static-score ties require the operating-point distinctions in Appendix C.5.

4.2 Localization and Low-FPR Detection

On identical held-out windows, ProTAL achieves 55.00% macro W3 recall, compared with 12.62% for Transition innovation and 6.04–7.98% for the other dynamic controls. Splices are detected most reliably, while isolated swaps and distant-frame replacements remain harder. Figure 3 retains the earlier comparison using different candidate pools; the paired evaluation confirms the advantage on common inputs.

Table 1 compares exact-edge W1 recall at empirical clean-FPR caps of 0.1%, 0.5%, and 1%. Each method uses one threshold per cap across all five families, preserving score ties. The scans use 38,340 common held-out clean window–edge observations and are distinct from frozen calibration-threshold evaluation. ProTAL leads at all three caps and in every family across the sampled positive budgets from 0.05% to 10% (Figure 4).

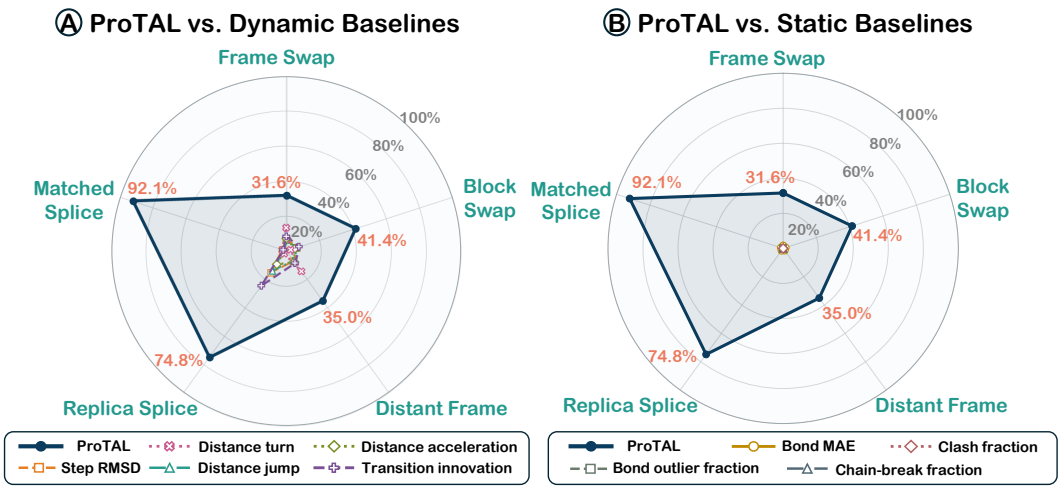


Figure 3: **W3 localization recall.** ProTAL is compared with **(A)** dynamic diagnostics, including reference-assisted Transition innovation, and **(B)** static controls at tie-safe operating points.

Table 1: **Exact-edge detection at low empirical false-positive rates.** All ten methods are evaluated using W1 recall under three clean-FPR caps. Actual attained FPRs are reported alongside five-family macro recall; all values are percentages.

Method	FPR cap 0.1%		FPR cap 0.5%		FPR cap 1.0%	
	Actual FPR (%)	Macro recall (%)	Actual FPR (%)	Macro recall (%)	Actual FPR (%)	Macro recall (%)
ProTAL	0.099	**29.75**	0.498	**46.64**	0.999	**51.53**
Cα bond-length MAE	0.099	0.09	0.496	0.48	0.999	0.97
Bond outlier fraction	0.000	0.00	0.000	0.00	0.000	0.00
Chain-break fraction	0.000	0.00	0.000	0.00	0.000	0.00
Clash fraction	0.000	0.00	0.000	0.00	0.000	0.00
Step RMSD	0.097	2.49	0.498	4.19	0.996	5.28
Distance jump	0.099	2.02	0.496	3.62	0.996	4.89
Distance acceleration	0.099	1.51	0.498	2.96	0.996	3.81
Distance turn	0.099	0.95	0.496	1.81	0.996	2.82
Transition innovation†	0.099	1.59	0.496	5.06	0.996	7.24

†Reference-assisted. FPR uses 38,340 common finite clean window–edge observations; recall is the unweighted five-family mean. Static fractions raise no alarms. Thresholds are scanned on held-out scores without interpolation or splitting ties (Appendix C.6).

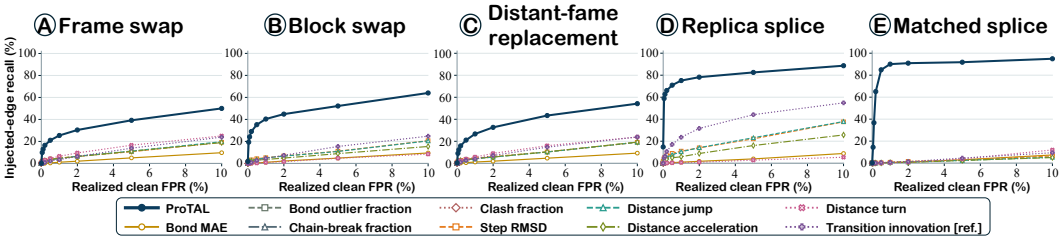


Figure 4: **Sensitivity versus realized clean FPR.** W1 recall for **(A)** frame swap, **(B)** block swap, **(C)** distant-frame replacement, **(D)** replica splice, and **(E)** matched splice. Markers are sampled held-out thresholds; lines guide the eye. All methods use common valid positions. Transition innovation is reference-assisted; static fraction controls overlap at the no-alarm point $(0, 0)$.

Table 2: W3 recall (%) on Test82 and the time-warp benchmark.

Method	Test82	Time warp
ProTAL	**56.20**	**3.18**
Geometry-TCN	22.70	2.43
Geometry-Transformer	19.20	2.74
AT (unsup.)	3.41	0.96
AT (sup.)	7.30	2.22
Conditional likelihood	7.15	2.10
Likelihood + sup. head	7.56	2.19
DCdetector	1.47	1.75
VAMPnet (adapted)	1.41	1.36
DADA (det.)	2.43	1.40

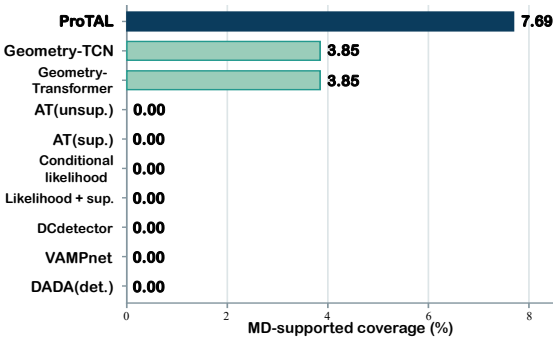


Figure 5: MD-supported coverage on cadence-adapted MDGen trajectories.

4.3 Comparison with Modern Temporal Detectors

We compare ProTAL with geometry-based TCN and Transformer controls Bai et al. (2018); Vaswani et al. (2017), adaptations of Anomaly Transformer (AT) Xu et al. (2022), DCdetector Yang et al. (2023), DADA Shentu et al. (2025), and VAMPnet Mardt et al. (2018), and conditional-likelihood controls (Table 2). Published methods are adapted following their official training or inference procedures, with changes for molecular inputs and transition scoring. Appendix C.2 reports the inputs, adaptations, actual training budgets, and checkpoint selection rules, distinguishing development-selected checkpoints from fixed final checkpoints and released weights.

To evaluate transfer to a perturbation family absent from ProTAL's training, we construct 820 control windows and 2,460 time-warped variants from the same 82 proteins. Smooth, monotonic changes to MD sampling times introduce local acceleration and deceleration while preserving frame order and window endpoints, without interpolating coordinates (Appendix C.7). ProTAL achieves 3.18% W3 recall, exceeding the strongest comparator, Geometry-Transformer, at 2.74%. Smooth time warping remains challenging for all evaluated methods. Matched clean-background rates and descriptive intervals for recall differences are reported in Appendix C.8.

We further evaluate 44 trajectories generated by a cadence-adapted MDGen model Jing et al. (2024b). With frozen detection thresholds, ProTAL achieves 7.69% MD-supported coverage (Appendix C.10), compared with 3.85% for both geometry baselines and 0% for the remaining methods (Figure 5). Appendix C.9 details trajectory generation and the overlap with MDGen's pretraining proteins.

4.4 MD Support and False-Alarm Behavior

We supplement constructed labels with conditional MD from 19 starting states across 18 proteins. Each state seeds 20 independent velocity realizations at 300 K, simulated for 1 ns, totaling 380 ns. Candidate endpoints receive tail support when their descriptive rank is at most 0.05 and the native endpoint control is not also flagged. Of 64 same-start candidates, 20 receive support, 42 are not rejected, and two remain unresolved. ProTAL detects 12 of the 20 supported cases (60%), versus nine (45%) for Transition innovation (Figure 6A). Coverage is measured within this selected set using position-conditioned endpoint compatibility (Appendix C.11).

The exact-edge audit on common inputs complements this evidence (Figure 6B). In the mixture containing clean windows once and each corruption set once, ProTAL obtains 42.96% recall and 46.19% FDP, compared with 3.24–9.64% recall and 76.45–88.18% FDP for the five dynamic comparators. Its clean-window FPR is 0.98% (95% protein-bootstrap interval: 0.79–1.18%). ProTAL therefore improves sensitivity and the proportion of label-consistent alarms at its calibrated operating point; these FDPs remain specific to the constructed-label mixture.

Table 3: Unique contribution to conditional-MD-supported case coverage. The denominator is 20 supported cases. The last column compares the union of all other methods before and after adding the row method. Static fractions use tie-safe thresholds.

Method	Unique support	Gain (pp)	Union: without → with
ProTAL	**5/20**	**+25**	**50% → 75%**
Transition innovation	2/20	+10	65% → 75%
Distance jump	0/20	+0	75% → 75%
Distance acceleration	0/20	+0	75% → 75%
Step RMSD	0/20	+0	75% → 75%
Bond MAE	1/20	+5	70% → 75%
Distance turn	0/20	+0	75% → 75%
Three static fractions (each)	0/20	+0	75% → 75%

4.5 Complementary Detection

ProTAL detects five conditionally MD-supported cases missed by every conventional comparator. Adding it increases their collective coverage from 50% to 75% of the 20 supported cases (Table 3). A separate analysis uses existing ATLAS replicas as an empirical reference, excluding every replica that contributes candidate frames. Among adjudicable, deduplicated exclusive alarms, descriptive mean MD-reference deviation rates favor ProTAL by 2.1–4.9 percentage points over five conventional controls, with variation across families (Figure 7A). Appendix C.15 reports the family-level estimates and reference-comparison protocol.

The `6irx_A` example illustrates the added coverage (Figure 7B–D). Replacing the native 32 ns frame with a 42 ns conformation preserves an MD-sampled structure but changes its temporal context. ProTAL flags both affected edges, while all six displayed comparators stay below threshold. Conditional MD from the 31 ns state assigns the constructed endpoint a discrepancy of 1.32 Å and tail rank $1/21$, versus 0.58 Å and $21/21$ for the native endpoint. These results support the detected endpoint mismatch (Appendix C.13).

4.6 Ablation Study

We evaluate six Test82 variants (Table 4). *Motion-only* and *Transitions-only* retain individual branches; *No-forecast* removes forecast-derived inputs; and *Single-scale* uses one motion-pooling scale. Geometry-TCN and Geometry-Transformer provide alternative temporal models.

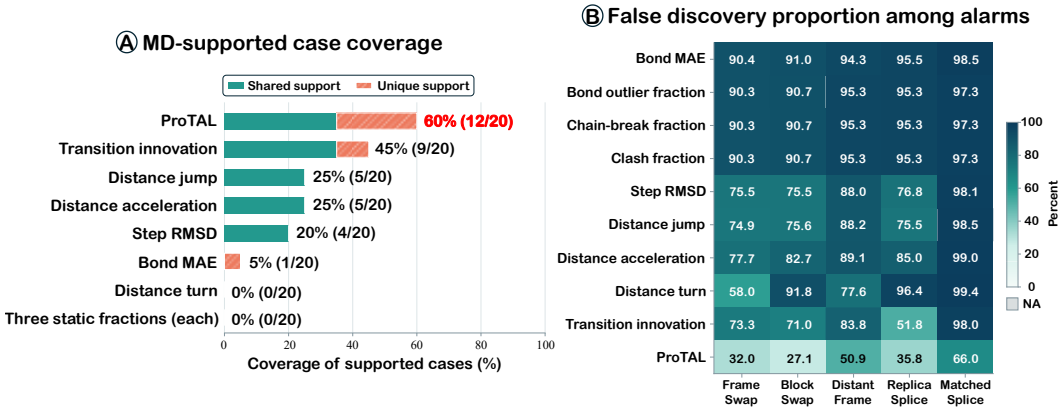


Figure 6: **MD support and alarm quality. (A)** Coverage of 20 conditionally MD-supported cases among 64 selected candidates; unique support means detection by only one displayed method. Static fractions use tie-safe thresholds. **(B)** Exact-edge FDP by corruption family under strict q99 thresholds. FDP concerns constructed labels and differs from clean FPR. Strict zero thresholds make the three static fractions alarm everywhere.

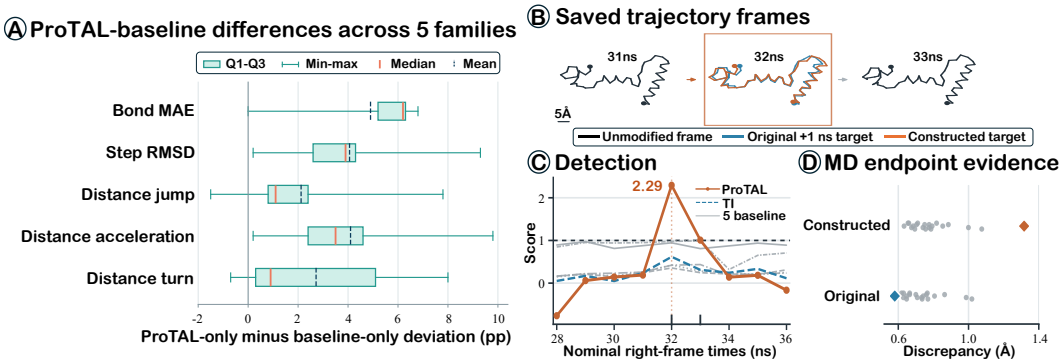


Figure 7: **Complementarity and a supported example. (A)** Difference in MD-reference deviation rates between ProTAL-only and baseline-only alarms (Δ_B, Appendix C.12); boxes summarize five family estimates, not confidence intervals. **(B)** A native 32 ns frame in `6irx_A` is replaced by a 42 ns frame. **(C)** Scores normalized by frozen alarm thresholds. **(D)** Constructed/native endpoint discrepancies (diamonds) against 20 conditional-MD endpoints (gray). Each row uses a separate 21-member comparison pool.

Table 4: Test82 ablations at fixed calibration thresholds (left) and matched 1% empirical test FPR (right). All values are percentages; W1 and W3 denote exact-edge and one-edge-tolerance recall. Best values are bold.

	Fixed calibration thresholds			Matched test FPR	
Method	W1 ↑	W3 ↑	FPR ↓	W1 ↑	W3 ↑
ProTAL	**51.87**	**55.49**	0.915	**52.42**	**56.20**
Transitions-only	46.74	50.86	1.023	46.46	50.53
Motion-only	19.71	23.05	0.920	20.08	23.68
No-forecast	46.13	50.50	1.013	45.98	50.34
Single-scale	46.33	50.01	0.996	46.36	50.05
Geometry-TCN	17.98	19.64	0.674	20.76	22.70
Geometry-Transformer	12.37	14.53	**0.561**	16.12	19.20

At fixed calibration thresholds (left), ProTAL reaches 51.87% W1 and 55.49% W3 recall, exceeding Transitions-only by 5.13 and 4.63 percentage points while lowering FPR from 1.023% to 0.915%. Fusion outperforms either branch, supporting complementary predictive and geometric evidence. The largest family-level gain occurs for matched splices (Appendix D.7). Removing forecast-derived inputs or using one pooling scale reduces W3 recall by 4.99 and 5.48 points, respectively, and increases FPR. At matched 1% empirical test FPR (right), ProTAL exceeds Transitions-only by 5.96 and 5.67 points in W1 and W3 recall. Its advantage therefore persists after matching false-alarm rates. Appendix D.8 provides results by perturbation family and across FPR budgets.

5 Conclusion

ProTAL localizes contextual anomalies by combining bidirectional latent prediction with residue-pair geometry. Controlled-corruption experiments show improved localization and low-FPR sensitivity, while selected MD studies support complementary detections. Evaluations on time warps absent from training and on MDGen outputs extend the assessment, although low time-warp recall leaves substantial room for improvement. These transition-level diagnostics complement structural and ensemble assessment. Their interpretation remains limited by synthetic supervision, partial Cα observations, and selected MD validation, motivating broader tests across perturbations.

Ethics Statement

This study uses publicly available protein simulation data and computationally generated trajectories. It does not involve human participants, animal experiments, or personally identifiable information. The source datasets, software, and pretrained models are acknowledged in the paper.

Reproducibility Statement

The code is available at the link: https://anonymous.4open.science/r/ProTAL-6EED.

Statement on AI Assistance

We used ChatGPT to assist with language editing, translation of the manuscript.

References

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15619–15629. IEEE, 2023.

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mido Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. Featured Certification.

David D Boehr, Dan McElheny, H Jane Dyson, and Peter E Wright. The dynamic energy landscape of dihydrofolate reductase catalysis. *science*, 313(5793):1638–1642, 2006.

Vincent B Chen, W Bryan Arendall, Jeffrey J Headd, Daniel A Keedy, Robert M Immormino, Gary J Kapral, Laura W Murray, Jane S Richardson, and David C Richardson. Molprobity: all-atom structure validation for macromolecular crystallography. *Biological crystallography*, 66(1): 12–21, 2010.

Katherine A Henzler-Wildman, Vu Thai, Ming Lei, Maria Ott, Magnus Wolf-Watz, Tim Fenn, Ed Pozharski, Mark A Wilson, Gregory A Petsko, Martin Karplus, et al. Intrinsic motions along an enzymatic reaction trajectory. *Nature*, 450(7171):838–844, 2007.

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

Bowen Jing, Bonnie Berger, and Tommi Jaakkola. AlphaFold meets flow matching for generating protein ensembles. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 22277–22303. PMLR, 21–27 Jul 2024a.

Bowen Jing, Hannes Stärk, Tommi Jaakkola, and Bonnie Berger. Generative modeling of molecular dynamics trajectories. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 40534–40564. Curran Associates, Inc., 2024b.

Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Foundations of Crystallography*, 32(5):922–923, 1976.

Siwon Kim, Kukjin Choi, Hyun-Soo Choi, Byunghan Lee, and Sungroh Yoon. Towards a rigorous evaluation of time-series anomaly detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 7194–7201, 2022.

Leon Klein, Andrew Foong, Tor Fjelde, Bruno Mlodozeniec, Marc Brockschmidt, Sebastian Nowozin, Frank Noe, and Ryota Tomioka. Timewarp: Transferable acceleration of molecular dynamics by learning time-coarsened dynamics. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 52863–52883. Curran Associates, Inc., 2023.

Sarah Lewis, Tim Hempel, José Jiménez-Luna, Michael Gastegger, Yu Xie, Andrew YK Foong, Victor García Satorras, Osama Abdin, Bastiaan S Veeling, Iryna Zaporozhets, et al. Scalable emulation of protein equilibrium ensembles with generative deep learning. *Science*, 389(6761): eadv9817, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Jiarui Lu, Bozitao Zhong, Zuobai Zhang, and Jian Tang. Str2str: A score-based framework for zero-shot protein conformation sampling. In *International Conference on Learning Representations*, volume 2024, pp. 10663–10694, 2024.

Jiarui Lu, Xiaoyin Chen, Stephen Lu, Chence Shi, Hongyu Guo, Yoshua Bengio, and Jian Tang. Structure language models for protein conformation generation. In *International Conference on Learning Representations*, volume 2025, pp. 98645–98678, 2025.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. Leworld-model: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026.

Andreas Mardt, Luca Pasquali, Hao Wu, and Frank Noé. Vampnets for deep learning of molecular kinetics. *Nature communications*, 9(1):5, 2018.

Frank Noé, Simon Olsson, Jonas Köhler, and Hao Wu. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147, 2019.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Jan-Hendrik Prinz, Hao Wu, Marco Sarich, Bettina Keller, Martin Senne, Martin Held, John D Chodera, Christof Schütte, and Frank Noé. Markov models of molecular kinetics: Generation and validation. *The Journal of chemical physics*, 134(17), 2011.

Mathias Schreiner, Ole Winther, and Simon Olsson. Implicit transfer operator learning: Multiple time-resolution models for molecular dynamics. *Advances in Neural Information Processing Systems*, 36:36449–36462, 2023.

Qichao Shentu, Beibu Li, Kai Zhao, Yang Shu, Zhongwen Rao, Lujia Pan, Bin Yang, and Chenjuan Guo. Towards a general time series anomaly detector with adaptive bottlenecks and dual adversarial decoders. In *International Conference on Learning Representations*, volume 2025, pp. 81358–81381, 2025.

Yann Vander Meersche, Gabriel Cretin, Aria Gheeraert, Jean-Christophe Gelly, and Tatiana Galochkina. Atlas: protein flexibility description from atomistic molecular dynamics simulations. *Nucleic acids research*, 52(D1):D384–D392, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Juan Viguera Diez, Mathias Schreiner, Ola Engkvist, and Simon Olsson. Boltzmann priors for implicit transfer operators. In *International Conference on Learning Representations*, volume 2025, pp. 38454–38482, 2025.

Christoph Wehmeyer and Frank Noé. Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics. *The Journal of chemical physics*, 148(24), 2018.

Jiehui Xu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Anomaly transformer: Time series anomaly detection with association discrepancy. In *International Conference on Learning Representations*, 2022.

Yiyuan Yang, Chaoli Zhang, Tian Zhou, Qingsong Wen, and Liang Sun. Dcdetector: Dual attention contrastive representation learning for time series anomaly detection. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 3033–3045, 2023.

A Architecture and Feature Definitions

The settings below describe the default configuration of our implementation. They specify a trajectory window with $T = 32$ sampled frames and $N = 64$ residue slots, including padding when needed. Invalid residues and pairs are masked. Parameters of the predictive encoder, temporal predictors, detection pathways, and fusion are initialized from scratch.

A.1 Frame Encoding and Directional Conditioning

For residue identities a, original sequence positions r, and validity mask m, frame encoding and projection give

$$z_t = P_\theta(E_\theta(X_t, a, r, m)) \in \mathbb{R}^d, \qquad d = 192, \tag{5}$$

Temporal context is introduced after frame encoding. The forward and backward forecasts are defined in Equation equation 1; their raw errors are

$$\begin{aligned} \epsilon^+_{i,h} &= d^{-1}\|\widehat{z}^+_{i+h|i} - \mathrm{sg}(z_{i+h})\|_2^2, \\ \epsilon^-_{i,h} &= d^{-1}\|\widehat{z}^-_{i|i+h} - \mathrm{sg}(z_i)\|_2^2. \end{aligned} \tag{6}$$

Stopping gradients through the residual baseline and target prevents the prediction loss from directly pulling those measurement coordinates together. The encoder still learns through predictor inputs and the clean representation objectives.

The GVP Encoder uses five message-passing layers with 128 scalar and 16 vector channels. Its graph construction selects at most 24 neighbors per residue, prioritizing local sequence neighbors, radius contacts, and remaining geometric neighbors. Node geometry includes local Cα distances, angle and torsion features, contact summaries, radial and distance statistics, and positional information; the initial vector channels encode tangent, normal, binormal, and radial directions. Edge descriptors combine 24 radial-basis channels with 30 geometric, sequence, and relation features. Amino-acid embeddings modulate geometric node features. Pooling includes scalar mean and maximum, four learned attention summaries, vector norms, and sequence-local variation, and produces a 128-dimensional frame representation. The projection head is

$$\mathrm{LN}_{128} \to \mathrm{Linear}_{128,192} \to \mathrm{LN}_{192} \to \mathrm{GELU} \to \mathrm{Linear}_{192,192}\,.$$

The scalar output is invariant to proper rigid rotations and translations. Signed geometric quantities retain chirality sensitivity; reflection invariance is not imposed.

Let p_t be a sinusoidal encoding of the physical timestamp, normalized by 1000 ps and scaled by 0.1, and let $\chi_t \in \mathbb{R}^2$ contain mean cosine and sine of valid Cα torsions. With boundary increments set to zero, define

$$v_t^+ = z_t - z_{t-1}, \qquad v_t^- = z_t - z_{t+1}, \tag{7}$$

$$u_t^\sigma = z_t + A([z_t; v_t^\sigma; \chi_t; p_t]) + p_t, \qquad \sigma \in \{+, -\}. \tag{8}$$

The shared adapter A is LN578–Linear578/192–GELU–Linear192/192. The backward Transformer processes $u^-_{T-1}, \ldots, u^-_0$; timestamps and chirality summaries are reversed in order without changing their signs. Physical time conditions the predictor and is excluded from its latent target. Each directional Transformer has four layers, width 192, six heads, feed-forward width 768, and zero dropout. Each attention layer permits the current token and seven preceding tokens in its own orientation. This is a per-layer mask: composition across layers can expand the effective context beyond eight input tokens. Horizon heads are affine 192-to-192 maps, initialized to zero so that predictions initially equal the source latent state.

A.2 Geometric Evidence and Scale-specific Integration

The 49-dimensional transition descriptor g_t contains three groups:

1. Forty-two differences of smooth distance-histogram responses: 14 Gaussian centers from 2 to 24 Å, bandwidth 2 Å, for crop-index separation bands 1–2, 3–8, and at least 9.
2. Two changes in the mean cosine and sine of valid Cα torsions.

3. Five pair-distance change statistics: signed mean, mean absolute change, root mean square, maximum absolute change, and fraction of pairs crossing an 8 Å contact threshold.

All valid non-self pairs contribute to the distance summaries. These fixed quantities use the observed coordinates, not a clean reference trajectory.

We project the heterogeneous evidence into a common feature space for each interval width $w \in \mathcal{W} = \{15, 7, 3, 1\}$. Let $e_t^{(w)}$ select both directional heads with $h \le (w+1)/2$; the width-one readout explicitly retains all eight heads. Define $G_t = [g_{t-2}; \ldots; g_{t+2}]$, with boundary replication. Integration takes the form

$$\begin{aligned} f_t^{(w)} = \phi_{E,w}\Big([\mathrm{LN}(e_t^{(w)}); \mathrm{mean}(e_t^{(w)}); \max(e_t^{(w)})]\Big) \\ + \phi_{\Delta}\Big([\mathrm{LN}(\Delta z_t); \log(1 + \|\Delta z_t\|_2/\sqrt{d})]\Big) \\ + \phi_G(\mathrm{LN}(G_t))\,, \end{aligned} \tag{9}$$

where all three terms are 64-dimensional. Explicit mean and maximum channels retain the absolute magnitude of predictive discrepancies alongside their normalized cross-head pattern. Thus, integration produces new transition features rather than passing the original frame embedding through unchanged.

Multi-scale Detector. For each width, we form a centered temporal summary using an equal mixture of elementwise mean and maximum pooling, denoted $\mathcal{P}_w$, and apply a scale-specific transformation ψ_w:

$$\begin{aligned} k_t^{(w)} &= \psi_w\Big(\mathcal{P}_w(f^{(w)})_t\Big)\,, \\ s_t^A &= b_A + w_A^\top[\bar{e}_t; k_t^{(15)}; k_t^{(7)}; k_t^{(3)}; k_t^{(1)}], \qquad \bar{e}_t = \tfrac{1}{8}\sum_{j=1}^{8} e_{t,j}. \end{aligned} \tag{10}$$

The final readout combines independent feature contexts from all four scales. Auxiliary coarse score maps support training, but do not gate the final score or enter it as averaged scores.

The predictive subsets contain 8, 6, 4, and 8 channels for widths 15, 7, 3, and 1, respectively. The discrepancy projector is affine; the latent-change and geometry projectors are two-layer GELU MLPs with 193 and 245 inputs and hidden/output width 64. Centered pooling uses replicate padding and

$$\mathcal{P}_w(f)_t = \tfrac{1}{2}\,\mathrm{mean}_{j\in\mathcal{N}_w(t)}\, f_j + \tfrac{1}{2}\,\max_{j\in\mathcal{N}_w(t)}\, f_j.$$

The scale transformation is LayerNorm–Linear64/64–GELU. Four 64-dimensional contexts plus the mean of eight predictive-error channels give 257 inputs to the Motion Predictor readout.

Auxiliary coarse hierarchy. Coarse widths 15, 7, and 3 have auxiliary score maps. If $k_t^{(w)}$ denotes an independent local context, the coarsest recursive context is $c_t^{(15)} = k_t^{(15)}$. For each subsequent coarse level with parent width p,

$$\gamma_t^{(w)} = 0.25 + 0.75\,\mathrm{sigmoid}\left(G_w([k_t^{(w)}; A_w c_t^{(p)}])\right), \tag{11}$$

$$c_t^{(w)} = k_t^{(w)} + \gamma_t^{(w)} A_w c_t^{(p)}, \tag{12}$$

$$s_t^{(w)} = \mathcal{P}_w(\mathrm{mean}\, e^{(w)})_t + 2\tanh\Big(H_w(c_t^{(w)})/2\Big)\,. \tag{13}$$

Here A_w, G_w, H_w are affine maps. The final Motion Predictor score uses the independent $k_t^{(w)}$, not these recursively combined contexts or the auxiliary score maps. Coarse world-model features are detached from their source computation, while the coarse projection and context networks remain trainable.

A.3 Complete Residue-pair Descriptor

The Transitions Reader constructs distances, increments, and fixed pair descriptors as

$$\begin{aligned} d_{t,ij} &= \|x_{t,i} - x_{t,j}\|_2, \qquad \delta_{t,ij} = d_{t+1,ij} - d_{t,ij}, \\ p_{t,ij} &= \Phi_{\mathrm{pair}}(d_{\cdot,ij}, i, j)_t \in \mathbb{R}^{32}. \end{aligned} \tag{14}$$

A 64-residue crop contains 2,016 unordered pair slots, with invalid pairs masked. The descriptor includes raw and normalized increments, deviations from preceding and following increments, soft-contact changes, endpoint distances, and crop-position separation.

For an unordered valid pair $i < j$, use the distances and increments in Equation 14. Distances are measured in Å. Normalize increments using their lower median over the candidate window:

$$\kappa_{ij} = \mathrm{median}_{\mathrm{lower},t}\,|\delta_{t,ij}| + 0.01\ \text{Å}, \tag{15}$$

$$n_{t,ij} = \mathrm{clip}(\delta_{t,ij}/\kappa_{ij}, -20, 20), \tag{16}$$

$$\rho_{t,ij} = \mathrm{clip}(\delta_{t,ij}/(0.10\ \text{Å}), -20, 20). \tag{17}$$

The lower median selects the lower middle value for an even count. It is computed separately for each pair within the observed window, rather than from population statistics. No time-step division is applied.

Suppress pair indices for clarity. Let $a_t^- = n_t - n_{t-1}$ and $a_t^+ = n_{t+1} - n_t$, replicating boundary values. For $w \in \{2, 4, 8\}$, let $\mu_{t,w}^-$ and $\mu_{t,w}^+$ average up to w preceding and following increments, excluding the current edge and clipping at the boundaries. An empty side uses n_t. Define $b_{t,w}^{\pm} = n_t - \mu_{t,w}^{\pm}$. Soft contact changes are

$$\Delta C_{t,c} = \mathrm{sigmoid}((c - d_{t+1})/0.5) - \mathrm{sigmoid}((c - d_t)/0.5), \qquad c \in \{8, 12, 16\},$$

with distances in Å. The descriptor concatenates the following groups in order:

$$\begin{aligned} p_{t,ij} = \big[&\rho_t, n_t, |n_t|, a_t^-, |a_t^-|, a_t^+, |a_t^+|; \\ &\{b_{t,w}^-, |b_{t,w}^-|, b_{t,w}^+, |b_{t,w}^+|\}_{w=2,4,8}; \\ &\{\Delta C_{t,c}, |\Delta C_{t,c}|\}_{c=8,12,16}; d_t/24, d_{t+1}/24; \eta_{ij}\big]. \end{aligned} \tag{18}$$

For crop-slot separation $s = j - i$, the five positional channels are

$$\eta_{ij} = [s/(N-1),\ \log(1+s)/\log N,\ \mathbf{1}[s \le 3],\ \mathbf{1}[4 \le s \le 7],\ \mathbf{1}[s \ge 8]]\,.$$

The group sizes are $7+12+6+2+5 = 32$. Crop-slot positions in this branch should not be confused with the original residue indices supplied to the GVP Encoder. The Transitions Reader receives only coordinates and the validity mask. Its distance-derived geometry is rigid-motion invariant, but cannot independently distinguish mirror images.

A.4 Residue and Temporal Networks

A shared MLP embeds each pair descriptor, with $v_{t,ji} = v_{t,ij}$ for the same unordered pair. For residue i, mean and maximum aggregation over its incident pairs produces a residue token, followed by a positional embedding. A residue Transformer models interactions among these tokens within each temporal edge. Its outputs are pooled into a transition representation:

$$\begin{aligned} v_{t,ij} &= \phi_{\mathrm{pair}}(p_{t,ij}), \\ r_{t,i} &= \phi_r([\mathrm{mean}_{j\neq i}\, v_{t,ij}; \max_{j\neq i} v_{t,ij}]) + \pi_i, \\ \widetilde{r}_{t,1:N} &= \mathrm{Transformer}_{\mathrm{res}}(r_{t,1:N}), \\ q_t^{(0)} &= \phi_{\mathrm{pool}}\left([\mathrm{mean}_i\, \widetilde{r}_{t,i}; \max_i \widetilde{r}_{t,i}; \sum_i \alpha_{t,i}\widetilde{r}_{t,i}]\right), \quad \alpha_{t,i} = \mathrm{softmax}_i(w_\alpha^\top \widetilde{r}_{t,i}). \end{aligned} \tag{19}$$

All aggregations respect the validity mask. Retaining residue tokens before global pooling allows the pathway to distinguish where geometric changes occur. The resulting $q_t^{(0)} \in \mathbb{R}^{128}$ describes a transition, rather than a frame-level latent state z_t.

2-way Multi-scale Patterns and Edge Scoring Head. We contextualize the transition sequence through four sequential residual temporal convolution blocks with dilations $(1, 2, 4, 8)$. Symmetric padding gives each block access to earlier and later transitions. With $Q^{(0)} = (q_0^{(0)}, \ldots, q_{T-2}^{(0)})$,

$$Q^{(\ell)} = Q^{(\ell-1)} + \mathcal{C}_{d_\ell}(Q^{(\ell-1)}), \qquad s_t^B = \mathrm{MLP}_{\mathrm{edge}}\Big(\mathrm{LN}(q_t^{(4)})\Big)\,. \tag{20}$$

Here $\mathcal{C}_d$ is a gated dilated convolutional block. In this module, *2-way* describes bidirectional temporal support, rather than a second pair of forecasting networks. The Edge Scoring Head produces one scalar for each of the $T-1$ transitions.

The pair MLP is LN32–Linear32/64–GELU–Linear64/32–GELU. Incident-pair mean and maximum embeddings are concatenated and projected to a 96-dimensional residue token. The added crop-position code is a learned linear projection of $[\xi_i, \sin(\pi\xi_i), \cos(\pi\xi_i), \sin(2\pi\xi_i)]$, where $\xi_i = i/(N-1)$ for zero-based slots. Residue Attention uses two pre-normalized Transformer layers with four heads, feed-forward width 384, GELU, dropout 0.1, and a final LayerNorm. Mean, maximum, and attention-weighted pooling produce 288 channels, projected by LN288–Linear288/128–GELU.

Each temporal residual block has the form

$$\mathcal{C}_d(Q) = \mathrm{Dropout}(\mathrm{Conv}_1[\mathrm{GLU}(\mathrm{Conv}_{3,d}(\mathrm{LN}(Q)))]) .$$

The kernel-three convolution maps 128 to 256 channels with symmetric zero padding of d; GLU returns 128 channels and the pointwise convolution preserves that width. The four dilations are sequentially $(1, 2, 4, 8)$. Their nominal convolutional receptive field is 31 transition tokens, but pair normalization and descriptor construction already use additional window context; 31 is not a claim about the entire branch's exact support. The Edge Scoring Head is LN128–Linear128/64–GELU–Linear64/1. Fusion is Linear2/1, initialized with weights $(1, 1)$ and zero bias. No sigmoid is applied to its output.

B Training Objectives and Optimization

This appendix specifies the default objective underlying Equation equation 4. All expectations are empirical means over the indicated training examples. Let $\rho(u)$ denote elementwise SmoothL1 with unit transition point, averaged over its arguments, and let $\ell_\tau(u) = \tau \log(1 + \exp(u/\tau))$. Clean-window representation objectives and anomaly-label objectives have separate roles: the former preserve predictive content, while the latter train localization of controlled corruptions.

Clean prediction objective. The mean component averages raw edge-aligned errors over clean windows:

$$\mathcal{L}_{\mathrm{pred}} = \frac{1}{2|\mathcal{H}|(T-1)} \sum_{\sigma\in\{+,-\}} \sum_{h\in\mathcal{H}} \sum_{t=0}^{T-2} E^{\sigma}_{t,h}, \tag{21}$$

Index the eight direction–horizon heads by $k = (\sigma, h)$. For a clean global minibatch of size B, define

$$\begin{aligned} m_k &= \frac{1}{B(T-1)} \sum_{b=1}^{B} \sum_{t=0}^{T-2} E^{\sigma}_{b,t,h}, \qquad \tau_h = 0.01, \\ \mathcal{L}_{\mathrm{clean}} &= \frac{0.5}{8} \sum_{k=1}^{8} m_k + 0.5\tau_h \log\left(\frac{1}{8} \sum_{k=1}^{8} e^{m_k/\tau_h} \right). \end{aligned} \tag{22}$$

The second term emphasizes heads with larger absolute prediction errors. Both terms use the edge-aligned errors of Equation equation 2; averaging unaligned prediction pairs would assign different boundary weights. Persistence errors are used to monitor predictive skill, not as an optimized denominator. Residual prediction heads initialize at zero. Targets and residual baselines are stop-gradient values from the current encoder, rather than outputs of a separate teacher network.

Clean predictive-error units. The nonlearned scales $\nu_{\sigma,h}$ in Equation equation 2 are estimated separately for the eight heads. Initial estimates and an evaluation-mode refit at the end of clean warmup use up to 2,048 clean training windows. For each head, the robust estimate is the maximum of median error, 1.4826 times median absolute deviation, and 10^{-6}. During training, let $u_{k,n}$ be this estimate from the current global minibatch's detached clean edge errors. Update

$$\begin{aligned} \nu_{k,n+1} &= \max\big(m_n \nu_{k,n} + (1-m_n) u_{k,n}, 10^{-6}\big) , \\ m_n &= \begin{cases} \max(0.95, n/(n+1)), & \text{clean warmup}, \\ 0.995, & \text{detection training}. \end{cases} \end{aligned} \tag{23}$$

where n is the number of previous updates. The updated buffer applies to the next forward pass. Tracking continues after warmup using only clean errors; evaluation and inference keep the saved scales fixed. The initial/refit median averages the middle two values for even sample counts, while minibatch updates use the lower middle value, matching their numerical implementations.

Clean representation auxiliaries. The unbudgeted auxiliary sum is

$$\begin{aligned}\mathcal{L}_{\text{aux}} = 0.10\mathcal{L}_{\text{tempdiv}} + 0.05\mathcal{L}_{\text{samplediv}} + 0.05\mathcal{L}_{\text{NCE}} \\ + \mathcal{L}_{\text{anchor}} + 0.10\mathcal{L}_{\text{anchor}\Delta} + 0.25\mathcal{L}_{\text{scale}}.\end{aligned} \tag{24}$$

For a matrix $U \in \mathbb{R}^{n\times d_U}$ of states or increments, center its columns to obtain U_c, set $v_j = \sqrt{n^{-1}\sum_i U_{c,ij}^2 + 10^{-6}}$, and normalize $\widehat{U}_{ij} = U_{c,ij}/\max(v_j, 10^{-4})$. Define

$$\mathcal{D}_{a,c}(U) = \frac{1}{d_U}\sum_j [a - v_j]_+ + \frac{c}{d_U}\left\|\text{offdiag}\left(\frac{\widehat{U}^\top\widehat{U}}{\max(n-1,1)}\right)\right\|_F^2. \tag{25}$$

$\mathcal{L}_{\text{tempdiv}}$ averages $\mathcal{D}_{0.1,0.1}$ on flattened temporal increments of encoder frame features and projected states. $\mathcal{L}_{\text{samplediv}}$ applies $\mathcal{D}_{0.2,0.05}$ across time-averaged projected states, one per clean window. These statistics combine the current distributed minibatch.

Temporal contrast uses only the one-step forecasts. For either temporal direction, let $\widehat{z}_{b,i}$ and $z^*_{b,i}$ denote the aligned prediction and correct content target. With $K = T - 1$ and cosine similarity sim,

$$\mathcal{L}^{\sigma}_{\text{NCE}} = -\frac{1}{BK}\sum_{b,i}\log\frac{\exp\Big(\text{sim}(\widehat{z}_{b,i}, \text{sg}(z^*_{b,i}))/0.1\Big)}{\sum_{j=1}^{K}\exp\Big(\text{sim}(\widehat{z}_{b,i}, \text{sg}(z^*_{b,j}))/0.1\Big)}, \qquad \mathcal{L}_{\text{NCE}} = \tfrac{1}{2}(\mathcal{L}^{+}_{\text{NCE}} + \mathcal{L}^{-}_{\text{NCE}}). \tag{26}$$

Negatives are other target times in the same clean window. This auxiliary does not contrast the two detection pathways.

For geometric anchoring, $u_t \in \mathbb{R}^{42}$ concatenates masked mean, standard deviation, and maximum over 14 fixed local geometric features. Fit per-channel median μ and scale $\gamma = \max(1.4826\,\text{MAD}, 10^{-3})$ using up to 2,048 clean training windows and freeze them. The target is

$$a_t = \text{clip}((u_t - \mu)/\gamma, -8, 8)\,R, \qquad R_{ij} = \frac{\sin(0.017ij)}{\sqrt{42}}, \quad 1 \le i \le 42,\ 1 \le j \le d. \tag{27}$$

Then $\mathcal{L}_{\text{anchor}} = \rho(z - a)$ and $\mathcal{L}_{\text{anchor}\Delta} = \rho(\Delta z - \Delta a)$. Let v^z, v^a be per-channel population standard deviations of centered temporal increments, computed with variance stabilizer 10^{-8}, with v^a detached. The scale floor is

$$\mathcal{L}_{\text{scale}} = \rho\big([\log(0.25v^a + 10^{-4}) - \log(v^z + 10^{-4})]_+\big). \tag{28}$$

This penalizes insufficient latent variation without shrinking larger variation. The auxiliary coefficient in Equation equation 4 is

$$\eta = \min\left(1, \frac{0.25\max(\text{sg}(\mathcal{L}_{\text{clean}}), 10^{-8})}{\max(\text{sg}(\mathcal{L}_{\text{aux}}), 10^{-12})}\right). \tag{29}$$

Thus the auxiliary loss contribution is limited relative to clean prediction, with no gradient through its budget. SIGReg is not included.

Family-balanced classification. For corruption family f, let $\mathcal{P}_f$ contain corrupted scores at original positive edge labels. Let $\mathcal{N}_f$ contain all paired-clean edge scores and corrupted scores outside the radius-one dilation of every positive label. The neighboring edges excluded from $\mathcal{N}_f$ are not automatically positive labels. For any score stream v,

$$\mathcal{L}^{\text{bal}}_{\text{BCE}}(v) = \frac{1}{5}\sum_{f=1}^{5}\left[\frac{1}{2}\mathbb{E}_{u\in\mathcal{P}_f}\log(1 + e^{-u}) + \frac{1}{2}\mathbb{E}_{u\in\mathcal{N}_f}\log(1 + e^{u})\right]. \tag{30}$$

Families and classes receive equal weight regardless of their edge counts. This loss supervises both s and s^B with coefficients 1 and 0.5, respectively. Scores are logits; no probability calibration is implied.

Quantiles and label-local pooling. For each Neyman–Pearson objective, standardize scores using the current global clean-score mean and population standard deviation, with the latter floored at 0.01. Both statistics remain differentiable. Write the standardized scores as v^c, v^b. The threshold has an exact empirical higher-quantile forward value and an implicit smooth backward derivative:

$$q = \mathrm{sg}(Q^{\mathrm{higher}}_{1-\alpha}(v^c)) + q_{\mathrm{soft}} - \mathrm{sg}(q_{\mathrm{soft}}), \qquad \frac{1}{|v^c|}\sum_j \mathrm{sigmoid}\left(\frac{v^c_j - q_{\mathrm{soft}}}{0.25}\right) = \alpha. \tag{31}$$

The root uses 24 bisection iterations; its derivative weights are proportional to $p_j(1-p_j)$, where p_j is the displayed sigmoid. The final score uses $\alpha = 0.01$ and only current-minibatch scores.

Define $\mathcal{M}_\tau(S)$ to have forward value $\max S$ and backward derivative of $\tau \log(|S|^{-1}\sum_{u\in S} e^{u/\tau})$. For every positive edge t, independently form

$$\begin{aligned} p_t &= \mathcal{M}_{0.20}\big(\{v^b_j : j \in \{t-1, t, t+1\} \cap [0, T-2]\}\big), \\ \widetilde{h}_t &= \mathrm{sigmoid}((p_t - q)/0.25), \\ h_t &= \mathrm{sg}(\mathbf{1}[p_t \geq q]) + \widetilde{h}_t - \mathrm{sg}(\widetilde{h}_t). \end{aligned} \tag{32}$$

Consecutive labels remain separate units. The forward hit is exactly the three-edge localization decision; only its derivative is smoothed.

Operating-point objective. Let $\mathcal{A}(\ell_1, \ldots, \ell_F)$ denote a family-robust reduction:

$$\mathcal{A}(\ell) = \tfrac{1}{2}\sum_f \omega_f \ell_f + \tfrac{1}{2}\sum_f \mathrm{sg}(\mathrm{softmax}_f(\ell/0.05))\,\ell_f, \tag{33}$$

where normalized ω_f are proportional to 1.75 for matched splicing, 1.35 for distant-frame replacement, and 1 for other families. Unlike Equation equation 30, this reduction emphasizes difficult families. Applied componentwise, the NP loss is

$$\mathcal{L}_{\mathrm{NP}} = \mathcal{L}_{\mathrm{tail}} + 0.5\mathcal{L}_{\mathrm{quant}} + \mathcal{L}_{\mathrm{rate}} + 0.5\mathcal{L}_{\mathrm{neg}} + \mathcal{L}_{\mathrm{group}}. \tag{34}$$

Let $\mathcal{T}_\alpha$ contain the largest $\lceil \alpha |v^c| \rceil$ clean scores. Tail ranking averages $\ell_{0.25}(u + 0.25 - p_t)$ over $u \in \mathcal{T}_\alpha$ and positives within each family, then applies $\mathcal{A}$. The quantile term applies $\mathcal{A}$ to $\ell_{0.25}(q + 0.25 - Q^{\mathrm{higher}}_{1-r_*}(p_f))$, with the same implicit quantile derivative; final target recall is $r_* = 0.70$.

Let R_μ and R_M denote micro and family-macro means of h_t. The weighted recall is $R_W = \sum_f \mathrm{sg}(\mathrm{softmax}_f(-R_f/0.025))R_f$. The rate loss is

$$\mathcal{L}_{\mathrm{rate}} = 0.20\ell_{0.05}(r_* - R_\mu) + 0.50\ell_{0.05}(r_* - R_M) + 0.30\ell_{0.05}(r_* - R_W). \tag{35}$$

The negative term applies $\mathcal{A}$ to the mean $\ell_{0.25}(v^b_j - q)$ over each family's top 5% corrupted scores outside its configured exclusion mask. Empty pools contribute no examples. For clean protein-entry/crop groups, let F_g be the mean sigmoid exceedance at temperature 0.25. The final-path group penalty is

$$\mathcal{L}_{\mathrm{group}} = 0.5\sum_g \mathrm{sg}(\mathrm{softmax}_g(F/0.01))\left[\frac{\ell_{0.01}(F_g - 0.03)}{0.05}\right]^2. \tag{36}$$

This is an optimization penalty, not a guarantee on group false-positive rates.

Paired and multiscale localization terms. For paired contrast, first compute $d_j = (s^b_j - s^c_j)/\sigma_c$, where σ_c is the differentiable clean-score standard deviation floored at 0.01. At each positive label, apply $\ell_{0.25}(0.25 - \mathcal{M}_{0.25}(\{d_j : j \in W_3(t)\}))$ and reduce by $\mathcal{A}$. Pooling paired differences is distinct from subtracting two separately pooled scores. Stability applies $\rho(d_j)$ outside the configured support mask. Halo ranking penalizes $\ell_{0.25}(p_{\mathrm{halo}} + 0.25 - p_{\mathrm{core}})$, using $\mathcal{M}_{0.25}$ pools. The halo excludes the union of all event-core neighborhoods. These terms also use family-robust reduction.

For widths $w = (15, 7, 3)$, auxiliary coarse NP losses use contiguous positive-event intervals expanded by $1 + \lfloor w/2 \rfloor$, with $(\alpha_w, r_w) = (0.10, 0.98), (0.05, 0.95), (0.03, 0.85)$. Their rate term uses sigmoid forward values, and their group term is zero. The final path instead uses Equation equation 32. Configured exclusion radii are $R_w = C + \max \mathcal{H}_w - 1 + \lfloor w/2 \rfloor$, using the per-layer

attention context $C = 8$, the selected horizons of each scale, and all horizons for width 1; final masks use $R = \max_w R_w$. These are training mask definitions, not a proof that dependencies of the full network vanish outside them. Direct BCE retains its separate radius-one exclusion.

With $(a_{15}, a_7, a_3) = (0.25, 0.35, 0.40)$, the hierarchy is

$$\begin{aligned} \mathcal{L}_{\text{hier}} &= \mathcal{L}_{\text{final}} + 0.5 \sum_{w \in \{15,7,3\}} a_w \mathcal{L}_w + 0.5\mathcal{L}_{\text{refine}}, \\ \mathcal{L}_{\text{final}} &= \mathcal{L}_{\text{NP}}(s) + 0.5\mathcal{L}_{\text{paired}} + 0.5\mathcal{L}_{\text{stable}} + 0.5\mathcal{L}_{\text{halo}}, \\ \mathcal{L}_w &= \mathcal{L}_{\text{NP}}(s^{(w)}) + 0.5\mathcal{L}_{\text{stable},w} + 0.5\mathcal{L}_{\text{halo},w}. \end{aligned} \tag{37}$$

Refinement compares the event-containing child pool against false sibling pools inside its parent, using $\ell_{0.25}(p_{\text{sibling}} + 0.25 - p_{\text{child}})$, for $15 \to 7$, $7 \to 3$, and $3 \to 1$. The three losses use weights $(0.25, 0.35, 0.40)$ after family reduction. Siblings overlapping any event's expanded child neighborhood are excluded. The last refinement uses the fused score. Width1 receives no duplicate coarse NP objective.

Optimization and gradient routing. The default schedule starts with eight clean-only epochs, followed by 20 epochs with detection enabled and exactly zero label gradient entering the Motion Predictor. Clean representation learning continues during this isolation phase. Subsequently, a forward-identity operator permits an adaptive backward multiplier b on detector gradients entering predictive features. Its maximum is 10^{-3}, probe reference 10^{-4}, and minimum useful value 10^{-7}. For measured label-to-clean latent-interface gradient ratio r at probe multiplier b_0, the initial budget is $\min(b_0, 10^{-3}, 0.8 \times 0.01\, b_0/r)$; invalid or zero-signal probes leave the bridge closed. Growth is capped by this measured budget and by clean-prediction checks.

Relative to the frozen clean-warmup reference, checks permit at most 10% aggregate and 20% per-head prediction-MSE increase, require positive fixed-denominator skill with a drop no greater than 0.08, and require at least half of the heads to outperform reference persistence. The geometry variation ratio must reach 0.25. Two consecutive safe validations allow multiplier growth by 2; failures reduce it by 0.5 and restore the predictive state. These controls protect the clean predictive objective during supervised adaptation.

AdamW uses learning rates 10^{-4} for the predictive model and 3×10^{-4} for detector parameters, with weight decay 10^{-3} and 0, respectively, and gradient clipping at 1. The predictive-model learning rate is multiplied by 0.25 after clean warmup. Detector learning rate is zero during clean warmup and then warms up over two epochs. The detection loss coefficient changes directly from 0 to 1 after clean warmup; it is not gradually ramped.

Calibration versus training thresholds. For the clean calibration set $\mathcal{D}_{\text{cal}}^{\text{clean}}$, inference uses

$$\theta = Q_{0.99}^{\text{higher}}\left(\{s_t(\mathcal{X}) : \mathcal{X} \in \mathcal{D}_{\text{cal}}^{\text{clean}}\}\right), \qquad \widehat{y}_t = \mathbf{1}[s_t \geq \theta]. \tag{38}$$

The current-minibatch quantiles above serve training objectives. Inference instead uses the fixed higher 99th percentile of raw clean calibration-edge scores in Equation equation 38, with $s_t \geq \theta$ including ties. Calibration protein entries are separate from training and held-out evaluation entries. Three-edge or whole-window pooling does not trigger threshold recalibration. Held-out edge, neighborhood, and window false-alarm rates are measured separately; none is guaranteed to equal 1% by this empirical calibration.

C Experimental Protocols and Additional Results

C.1 Benchmark Construction and Optimization

We construct the benchmark from ATLAS MD trajectories Vander Meersche et al. (2024), using protein-entry-disjoint subsets for training and evaluation. The training subset contains 200 proteins, 600 replica trajectories, and 12,780 windows. A separate 39-protein development subset is partitioned into 20 proteins for calibration and checkpoint selection (1,098 windows) and 19 held-out proteins for evaluation (1,278 windows). Each input contains 32 frames sampled at 1 ns intervals

and a crop of at most 64 Cα atoms; padding is masked. ProTAL outputs 31 adjacent-frame transition scores per window. Comparisons with conventional diagnostics and the associated MD analyses of constructed corruptions use the 19-protein evaluation subset. Figure 3 additionally retains historical baseline results from a separate training-corruption audit pool. Test82 is used for learned-detector comparisons and ablations, and supplies the windows for the time-warp challenge. The generated-trajectory evaluation uses a separate 11-protein cohort described in Appendix C.9.

We use five families: frame swapping, block swapping, distant-frame replacement, replica splicing, and matched splicing. These introduce errors in temporal order or continuity while drawing constituent conformations from MD trajectories. Replica donors come from another replica of the same protein and residue crop. Matched splicing additionally constrains boundary compatibility, making a large instantaneous structural change an insufficient detection criterion. The configured corruption fraction is 0.7, the block length is three frames, and the minimum separation parameter is seven frames. Corruption generation uses separate seeds for training, calibration, and evaluation. Positive labels identify affected temporal edges, so performance measures localization of constructed inconsistencies rather than physical invalidity of individual structures.

We train ProTAL from scratch for 188 epochs and use the checkpoint at epoch 132, selected by a calibration-set ranking that prioritizes W3 micro-recall and then macro-recall. Training uses separate AdamW optimizers for the predictive representation and detector, with initial learning rates of 10^{-4} and 3×10^{-4}, weight decays of 10^{-3} and zero, and a global batch of 64 windows. An eight-epoch clean warm-up precedes 20 epochs of detector isolation, followed by guarded adaptation. The detection-loss coefficient changes directly from zero to one after warm-up; the detector learning rate warms up separately over two epochs. The frame encoder has five message-passing layers, 128 scalar channels, and 16 vector channels; the latent width is 192. Each directional predictor has four Transformer layers and six attention heads, with prediction horizons $\{1, 2, 4, 8\}$. The model architecture and training objectives are specified in the method section and its appendices. All reported detections use the same frozen checkpoint.

C.2 Learned-baseline Implementations

Table 5 documents the configurations used in Table 2. Checkpoint selection follows the method-specific rules listed below. Clean-dev denotes uncorrupted development data, and NLL denotes negative log-likelihood.

C.3 Threshold Protocol for Learned-Detector Comparisons

Table 2 uses retrospective clean-edge FPR budget matching separately on Test82 and the time-warp benchmark, with model weights fixed. Thresholds use only the corresponding dataset's clean scores, without using anomaly labels or optimizing recall or F1. The common budget is $q = 2137/213714$, reported as 1%, with at most $K = \lfloor qN \rfloor$ false alarms among N clean edges. Test82 has $N = 213{,}714$ and $K = 2{,}137$; the time-warp controls have $N = 25{,}420$ and $K = 254$. For each method and dataset, let $s_{(1)} \le \cdots \le s_{(N)}$ denote the clean scores sorted in float64. We set

$$\theta = \mathrm{nextafter}_{64}\big(s_{(N-K)}, +\infty\big), \qquad \mathrm{alarm}(s) = \mathbf{1}[s \ge \theta],$$

where $\mathrm{nextafter}_{64}$ returns the next representable float64 value toward $+\infty$. Ties at the cutoff are excluded together, without random splitting or interpolation, so realized FPR may fall below the budget. Each method uses one threshold across all five Test82 corruption families and a separate threshold across all three time-warp amplitudes, without protein-specific, family-specific, or amplitude-specific adjustment. Actual FPR is FP/N on clean edges. W3 permits a ± 1-edge match within the same window only when computing recall; it does not trigger threshold recalibration. These retrospective operating points differ from the frozen clean-calibration thresholds used elsewhere and do not establish an independently calibrated deployment FPR guarantee.

C.4 Model Configuration

Table 6 lists the architecture used for all reported ProTAL results. Inputs contain 32 frames at 1 ns intervals and up to 64 Cα atoms, with invalid residues and pairs masked. The encoder produces 192-dimensional latents for independent forward and backward predictors. A multi-scale readout

Table 5: Inputs, implementations, training budgets, and checkpoint rules for the learned comparisons. Update counts describe the actual runs; batch sizes and architectures differ across methods.

Method	Inputs and implementation	Actual training budget	Checkpoint rule
ProTAL	Cα graphs, residue and time information, and pairwise changes; GVP encoder with prediction and detection branches.	188 epochs; 33,276 global steps, including 31,860 detector-supervised updates.	Calibration W3 ranking; epoch 132.
Geometry-TCN	8,117-dimensional geometric features per edge; 4-layer TCN with width 192.	230,040 steps; two GPUs; 10 paired windows per step.	Fixed epoch 188.
Geometry-Transformer	The same 8,117-dimensional geometric features; 4 layers, width 192, and 6 attention heads.	230,040 steps.	Fixed epoch 188.
AT (unsup.)	Local adaptation using 8,068-dimensional edge features; 3 layers, width 128, and 4 heads.	6,000 steps; batch size 32.	Minimum clean-dev reconstruction loss; step 6,000.
AT (sup.)	AT backbone with a supervised edge-classification head.	6,000 steps.	Maximum development macro-W3; step 5,000.
Conditional likelihood	Distance-increment modeling with a 3-layer GRU and a 4-component Gaussian mixture.	6,000 steps.	Minimum clean-dev NLL; step 2,500.
Likelihood + sup. head	Frozen conditional-density model with a supervised detection head.	6,000 density-model steps plus 6,000 head-training steps.	Density: step 2,500; head: step 1,000.
DCdetector	Normalized distance channels; dual-attention detection adaptation.	6,000 steps; effective batch size 32.	Fixed final checkpoint; step 6,000.
VAMPnet (adapted)	Eight soft states; VAMP2 training with an added transition-anomaly score.	6,000 steps, plus transition counts fitted on clean training data.	Maximum clean-dev VAMP2; step 6,000.
DADA (det.)	Frozen official weights; sequence-length adaptation and variance-based scoring across 10 reconstructions.	No local training; external pretraining budget unavailable.	Fixed official weights.

combines predictive errors with latent and geometric changes; a separate residue-pair pathway provides a complementary score. Learned affine fusion produces 31 raw transition scores per window. Inference requires no reference trajectory.

The attention window of eight tokens is a per-layer mask, so stacking four layers can expand the predictor's effective context. Absolute timestamps are normalized by 1,000 ps and encoded sinusoidally with scale 0.1. The latent-change and geometric-context MLPs use LayerNorm followed by two linear layers with an intervening GELU. Geometric context concatenates five neighboring 49-dimensional transition descriptors. Prediction targets use detached states of the current encoder. The horizon heads are initialized to zero, making the initial forecasts equal to their source latent states. The motion score readout initially selects the mean normalized prediction error; the remaining readout weights and its bias are zero. All other learned modules are initialized from scratch.

Training uses separate AdamW optimizers for the predictive representation and detection modules, as detailed in Table 7. Eight clean-only epochs precede detection training. From epoch 9, the detection-loss coefficient is set directly to one; only the detector learning rate is warmed up over two epochs. During the first 20 detection epochs (epochs 9–28), label gradients do not enter the predictive representation, which continues to learn from clean trajectories. Subsequent feedback is

Table 6: **ProTAL architecture.** Dimension sequences specify consecutive linear-layer widths. GVP denotes the scalar–vector graph encoder; FFN denotes a Transformer feed-forward network.

Component	Configuration
Input and frame encoding	
Input window	32 frames; at most 64 residue slots; 1 ns frame spacing
GVP encoder	5 message-passing layers; 128 scalar and 16 vector channels; dropout 0
Residue graph	At most 24 neighbors; sequence span 4 and contact radius 12 Å guide neighbor priority
Distance features	24 radial-basis channels; maximum radial-basis distance 24 Å
Identity and pooling	Amino-acid embedding width 32; 4 learned attention pooling queries; frame feature width 128
Latent projection	$128 \to 192 \to 192$, with LayerNorm and GELU
Bidirectional prediction and multi-scale detection	
Conditioning adapter	$578 \to 192 \to 192$; inputs combine latent state, directional increment, chirality and time encoding
Directional predictors	2 independent Transformers; each has 4 layers, width 192, 6 heads, FFN width 768 and dropout 0
Attention and horizons	Causal window of 8 tokens per layer and direction; horizons $\{1, 2, 4, 8\}$; each horizon head is $192 \to 192$
Interval scales	Widths $(15, 7, 3, 1)$; 64-dimensional evidence and context at each scale; equal mean/max pooling weights
Discrepancy projection	$(8, 6, 4, 8)$ selected error channels for the four scales, respectively; append their mean and maximum, then project to 64
Additional evidence	Latent-change MLP: $193 \to 64 \to 64$; geometric-context MLP: $245 \to 64 \to 64$
Motion score readout	Affine $257 \to 1$: four independent 64-dimensional scale contexts and the mean of eight normalized prediction errors
Residue-pair pathway and score fusion	
Pair embedding	32-dimensional descriptor; MLP $32 \to 64 \to 32$ with GELU
Residue attention	Token width 96; 2 Transformer layers; 4 heads; FFN width 384; dropout 0.1
Residue pooling	Mean, maximum and attention-weighted summaries: $288 \to 128$
Temporal network	4 residual convolutional blocks; width 128; kernel 3 with symmetric padding; dilations $(1, 2, 4, 8)$; GLU; dropout 0.1
Geometric score head	$128 \to 64 \to 1$, with LayerNorm and GELU
Score fusion	Affine $2 \to 1$; initial weights $(1, 1)$ and bias 0; no output sigmoid

Table 7: **Optimization configuration.** Predictive and detector learning rates follow separate schedules. The gradient multiplier changes backward propagation into predictive features without changing their forward values.

Setting	Value
Initialization and seed	From scratch;
Global batch	64 base trajectory windows, plus their paired corruptions during detection training; no gradient accumulation
Training and selection	188 epochs; regular calibration validation every 4 epochs; selected checkpoint at epoch 132
Predictive optimizer	AdamW; base learning rate 10^{-4}; weight decay 10^{-3}
Detector optimizer	AdamW; base learning rate 3×10^{-4}; weight decay 0
Gradient clipping	Norm limit 1.0, applied separately to the two parameter groups
Predictive LR schedule	3-epoch linear warm-up, followed by cosine decay to 5% of the base rate; additional factor 0.25 after clean warm-up and adaptive safety reductions
Detector LR schedule	Zero during clean warm-up; 2-epoch warm-up and constant rate during the initial isolation stage; subsequent cosine decay to 5% of the base rate
Clean warm-up	Epochs 1–8; detection loss disabled
Detection activation	Detection-loss coefficient 1 from epoch 9; no gradual loss ramp
Gradient isolation	20 detection epochs with zero label-gradient feedback to predictive features
Adaptive gradient feedback	Maximum multiplier 10^{-3}; probe reference 10^{-4}; minimum useful multiplier 10^{-7}
Feedback controller	Growth factor 2 after two consecutive safe validations; backoff factor 0.5; feedback remains subject to clean-model checks

enabled only when clean-prediction checks permit it and uses an adaptive gradient multiplier that may remain zero. The reported model is the epoch-132 checkpoint selected from a 188-epoch run using the calibration-set hard-recall ranking, with W3 micro-recall and macro-recall as the leading criteria. The same frozen parameters are used in fixed-threshold evaluations and empirical FPR sweeps.

C.5 Operating Points and Evaluation Units

Static controls comprise Bond MAE, Bond outlier fraction, Chain-break fraction, and Clash fraction. These Cα-level proxies are motivated by structural validation Chen et al. (2010) and do not implement the full MolProbity suite. Local dynamic controls comprise Step RMSD after optimal rigid alignment Kabsch (1976), Distance jump,Distance acceleration, and Distance turn. Transition innovation additionally uses other clean replicas of the same protein and is therefore reference-assisted. ProTAL requires no reference trajectory at inference. Continuous-score operating points use the higher empirical 99th percentile of clean calibration-edge scores, with alarms defined by $s_t \geq \theta$; ProTAL's frozen threshold is 1.64847. For discrete static scores, this rule can yield a zero threshold and an all-alarm detector. We therefore distinguish the strict rule in Figure 6B from the tie-safe static controls used in Figures 3 and 6A and Table 3. These are specified operating points, not a guarantee of equal held-out false-positive rates.

W3 recall counts a labeled edge as detected when an alarm occurs at that edge or either immediate neighbor. Consecutive positive labels remain separate evaluation units. We report per-family recall and the unweighted mean across the five families. For the false-alarm audit, we instead use exact-edge labels (W1) and positions where all methods have finite scores, giving 30 shared positions per window. We distinguish the false-positive rate, $\mathrm{FPR} = \mathrm{FP}/(\mathrm{FP} + \mathrm{TN})$, from the empirical false-discovery proportion among alarms, $\mathrm{FDP} = \mathrm{FP}/(\mathrm{TP} + \mathrm{FP})$. The latter is undefined when no alarms are raised. Figure 3 retains a historical comparison with protocol-aligned but different candidate pools for ProTAL and the baselines; the paired follow-up and subsequent audits use common scored inputs.

C.6 Empirical Threshold Scans

We assess whether ProTAL retains its detection advantage when false alarms are constrained across a range of operating points. Figure 4 plots exact-edge injected-corruption recall (W1) against realized FPR on unmodified clean windows, using saved held-out scores. The common finite-score mask yields 38,340 clean window–edge observations. We scan nine empirical FPR budgets from 0% to 10%. For each method and budget, we select the lowest threshold satisfying the clean-FPR cap among distinct pooled clean and positive scores, or a no-alarm point if none qualifies. This threshold is shared across all five families and preserves tied-score groups. The horizontal coordinate is the achieved FPR, rather than the requested budget. Counts refer to window–edge occurrences, which need not be statistically independent.

Actual FPR is $100\,\mathrm{FP}/N_{\mathrm{clean}}$ on common finite held-out clean window–edge observations ($N_{\mathrm{clean}} = 38{,}340$), not the false-discovery proportion in a constructed mixture. Macro recall equally weights frame swap, block swap, distant-frame replacement, replica splice, and matched splice; it is neither pooled recall nor a per-protein average. Each method uses one threshold per cap, shared across all five families. Attainable FPRs are reported without interpolation or splitting tied scores. The three static fraction controls have valid no-alarm operating points at all listed caps. Transition innovation is reference-assisted; Cα bond-length MAE is denoted Bond MAE elsewhere. These held-out threshold scans do not provide independently calibrated deployment FPR guarantees. Bold marks the highest unrounded macro recall per column, including any exact ties; no statistical significance is asserted.

C.7 Construction of the Time-Warp Benchmark

We use ATLAS trajectories saved at 10-ps intervals for the 82 Test82 proteins. Each window contains 32 frames at nominal 1-ns intervals, spanning 31 ns, and retains the original Cα residue crops and padding masks.

For each window, we construct three variants with amplitudes $a \in \{0.25, 0.50, 0.75\}$. Within an interval $[t_0, t_0 + L]$, source sampling times follow

$$\phi(t) = t + \frac{saL}{C} \sin^4\left(\pi \frac{t - t_0}{L}\right), \qquad L = 12\,\mathrm{ns}, \quad C = \frac{3\sqrt{3}\pi}{4}, \tag{39}$$

with $\phi(t) = t$ outside this interval. The onset t_0 and sign $s \in \{-1, +1\}$ are determined by the same hash-based procedure and shared across amplitudes. The mapping satisfies $1 - a \le \phi'(t) \le 1 + a$, preserving temporal order and window endpoints while introducing local acceleration and deceleration.

Transformed times are rounded to the nearest existing MD frame, with at most approximately 5 ps of quantization error. We use no coordinate interpolation, repeated or reversed source frames, or additional MD simulation. Detector inputs retain nominal 1-ns timestamps; actual source timestamps are recorded separately. The benchmark contains 820 unmodified controls and 2,460 time-warped windows. An edge is labeled as perturbed if either constituent source frame differs from its unmodified counterpart. These labels describe altered temporal parameterization, not physically impossible conformations. Time warping is absent from ProTAL's training; however, reuse of previously evaluated Test82 proteins makes this a test of transfer to an unseen perturbation family, rather than an independent blind test on new proteins.

C.8 Time-Warp Background Alarms and Recall Differences

We evaluate background alarms on the 820 unmodified control windows paired with the time-warp benchmark, using the dataset-specific threshold protocol in Appendix C.3. These controls contain 25,420 adjacent-frame edges. ProTAL flags 254 clean edges, giving an empirical clean-edge FPR of approximately 1%.

We distinguish two W3 background measures. The all-center background counts a clean edge position as a hit when an alarm occurs at that position or either valid neighboring edge within the same window. This gives 699 hits among 25,420 clean centers. The matched background instead evaluates the clean counterparts at the same target-edge positions used for time-warp recall, apply-

Table 8: Time-warp recall, clean-control background, and recall differences. Rates are percentages; differences and their descriptive 95% intervals are in percentage points.

Quantity	Estimate	95% interval
Rates (%)		
Clean-edge FPR (254/25,420)	1.00	—
All-center clean W3 (699/25,420)	2.7498	—
Matched clean W3 background	2.8984	—
Time-warp macro W3 recall	3.1843	—
Differences (percentage points)		
Time warp minus matched background	+0.2859	$[-0.4639,\ 1.0156]$
ProTAL minus Geometry-TCN	+0.758	$[0.128,\ 1.378]$
ProTAL minus Geometry-Transformer	+0.444	$[-0.270,\ 1.114]$

ing the same W3 matching rule and aggregation. This retains the target-position distribution when comparing perturbed windows with their clean controls.

Let R_{warp} denote macro W3 recall on perturbed windows and B_{paired} the corresponding matched clean background. The perturbation-associated increment is

$$\Delta_{\mathrm{background}} = R_{\mathrm{warp}} - B_{\mathrm{paired}}. \tag{40}$$

For a comparator m, we separately report the raw recall difference $\Delta_m = R_{\mathrm{ProTAL}} - R_m$. These between-method differences are calculated before background subtraction. Table 8 reports the rates, differences, and descriptive 95% intervals.

C.9 Trajectory Generation with Cadence-Adapted MDGen

We use MDGen-200ps, a local adaptation of the officially released MDGen ATLAS checkpoint Jing et al. (2024b). Its sampling cadence is adapted from 400 ps to 200 ps using timestamp-verified ATLAS trajectories from 16 training proteins and four separate validation proteins. All 11 evaluation proteins are excluded from this additional adaptation. We perform 512 AdamW Loshchilov & Hutter (2019) updates with learning rate 10^{-5} and select the epoch-7 checkpoint (448 updates) by the lowest mean validation flow-matching loss, independently of downstream detection performance.

With this frozen checkpoint, we generate four trajectories per evaluation protein using 4 seeds, yielding 44 trajectories. Each trajectory starts from an initial protein conformation and comprises two consecutive 250-frame blocks; the final conformation of the first conditions the second. Removing the duplicated first frame of the second block yields 499 frames at nominal 200-ps intervals. For detection, we retain frame indices $0, 5, \ldots, 465$, giving 94 frames spanning 0–93 ns at nominal 1-ns intervals. This produces 93 transitions per trajectory and 4,092 transitions overall, including transitions across the generation-block boundary.

No coordinate interpolation, boundary smoothing, or additional MD simulation is used during generation. The assigned intervals reflect the generator's learned sampling cadence rather than explicit MD integration. Although excluded from the additional adaptation, 10 of the 11 evaluation proteins occur in MDGen's original pretraining split. This experiment therefore evaluates detection on generated trajectories, rather than strictly unseen-protein generalization.

C.10 MD-Reference Adjudication and Coverage

We evaluate all adjacent transitions from the fixed 11-protein cohort described in Appendix C.9. Each generation seed contributes 1,023 candidate edges, giving 4,092 edges across seeds 42–45. Within this cohort, candidate edges are included regardless of detector alarms. An automated MD-reference procedure uses the R3 reference trajectories for fitting, calibration, and applicability checks, assigning reference-supported deviation labels independently of detector outputs.

Of the 4,092 candidate edges, 26 receive MD-reference support for a deviation, while 3,694 have insufficient reference evidence and remain unresolved. Unresolved edges are not assigned normal labels or counted as missed supported deviations. Let N_{MD} denote the number of reference-supported

Table 9: Counts underlying ProTAL's coverage of MD-reference-supported deviations on generated trajectories.

Generation seeds	Candidate edges	Supported	Detected	Coverage (%)
42 (historical)	1,023	8	1	12.50
43	1,023	6	0	0.00
44	1,023	7	0	0.00
45	1,023	5	1	20.00
43–45 (primary)	3,069	18	1	5.56
42–45 (secondary pooled)	4,092	26	2	7.69

deviations and n_m the number detected by method m under its frozen detection threshold. We report

$$\text{Coverage}_m = 100\,\frac{n_m}{N_{\text{MD}}}\%. \tag{41}$$

The denominator therefore counts supported deviations identified through reference adjudication, rather than candidates selected from detector alarms.

Table 9 provides the counts underlying ProTAL's coverage. Seed 42 belongs to the historical evaluation. The original primary analysis uses seeds 43–45, yielding one detection among 18 supported deviations (5.56%). The secondary pooled analysis includes seeds 42–45, yielding two detections among 26 supported deviations (7.69%), as reported in Figure 5. The pooled percentage is calculated from summed counts, rather than by averaging seed-level percentages.

C.11 Conditional-MD Protocol and Case Coverage

We assess whether selected suspicious transitions are also unusual under newly simulated dynamics. The case study spans 19 full-system starting states from 18 proteins, selected across corruption families and ProTAL/Transition-innovation alarm strata. From each starting configuration, we draw 20 independent velocity realizations at 300 K and simulate 1 ns with the original all-atom topology and simulation settings, using a 2 fs time step. This yields 380 branches and 380 ns of aggregate simulation. Candidate endpoints are compared with the simulated endpoint ensemble through 1,830 nonlocal Cα pair distances. A descriptive endpoint rank at most 0.05 provides tail evidence, provided that the corresponding native endpoint control is not also flagged. This tests position-conditioned endpoint compatibility, rather than replaying the original trajectory or establishing a full-path likelihood.

The same-start expansion contains 64 candidate transitions, of which 20 receive MD tail support, 42 are not rejected, and two remain unresolved because of the native control. Figure 6A reports coverage of the 20 supported cases. ProTAL detects 12 (60%), compared with nine (45%) for Transition innovation, five each (25%) for Distance jump and Distance acceleration, four (20%) for Step RMSD, and one (5%) for Bond MAE. Distance turn and the tie-safe static fraction controls detect none. Because case selection is enriched by detector agreement and disagreement, and several candidates share a starting state, these percentages describe the selected case study rather than population recall. Nevertheless, they show that a substantial subset of ProTAL's detections has additional support from conditional MD.

We next ask whether ProTAL provides evidence beyond detecting more of the same cases. Table 3 measures each method's unique contribution to the conditional-MD-supported set: a case is unique to a method if that method detects it and none of the other listed methods does. ProTAL contributes five such cases, or 25 percentage points of the 20-case denominator. Adding ProTAL to all conventional comparators increases their collective coverage from 50% to 75%, corresponding to five additional supported cases. Transition innovation and Bond MAE contribute two and one globally unique cases, respectively; the remaining controls contribute none. ProTAL therefore provides the largest unique contribution in this selected set, while the nonzero contributions of other methods also support their complementary use.

C.12 MD-REFERENCE ADJUDICATION AND EXCLUSIVE ALARMS

Figure 7A provides a separate assessment of exclusive alarms using the existing ATLAS MD replicas as an empirical reference. Every replica contributing frames to a candidate window, including splice donors, is excluded from its reference. We use two reversed library/calibration folds: the two remaining replicas for single-source candidates, or two temporally separated blocks of the remaining replica for two-source candidates. Candidate transitions are compared with the native 1 ns successors of their five nearest reference starting conformations. A transition is reference-deviant only when its initial state is covered, its transition lies in the empirical upper 5% tail in both folds, and native-control quality checks pass. Conflicting or insufficient evidence is marked uncertain and excluded from the adjudicable denominator.

For a baseline B, let A_P and A_B denote the exact-edge alarm sets of ProTAL and B, and let D and J denote reference-deviant and adjudicable transitions. We compare exclusive-alarm enrichment using

$$\Delta_B = 100 \left[\frac{|(A_P \setminus A_B) \cap D|}{|(A_P \setminus A_B) \cap J|} - \frac{|(A_B \setminus A_P) \cap D|}{|(A_B \setminus A_P) \cap J|} \right]. \tag{42}$$

Physical transitions are deduplicated within each family, and zero-denominator comparisons remain undefined. Positive Δ_B indicates a higher fraction of MD-reference deviations among ProTAL-only alarms; it does not measure the number of unique detections.

Across the five corruption families, the displayed mean differences favor ProTAL by approximately 4.9 percentage points over Bond MAE, 4.1 over Step RMSD, 2.1 over Distance jump, 4.1 over Distance acceleration, and 2.7 over Distance turn. The boxes summarize five family-level point estimates, not confidence intervals; some family-level differences are negative, and the results do not establish a significant advantage for every comparison. Read together, Figure 7A and Table 3 provide two distinct kinds of evidence: ProTAL-exclusive alarms can be enriched for deviations from an empirical MD reference, and ProTAL adds conditionally MD-supported cases missed by all evaluated conventional diagnostics.

C.13 DETAILED CASE ANALYSIS

Figure 7B–D illustrates one of the five cases unique to ProTAL in Table 3: a distant-frame replacement in `6irx_A`. A conformation from 42 ns of the same replica replaces the frame at nominal time 32 ns, while the adjacent 31 and 33 ns frames are retained (panel B). Although the inserted structure is itself sampled from MD, its temporal placement creates two affected transitions. ProTAL crosses its alarm threshold at both edges, with normalized scores of 2.29 and 1.01 at nominal right-frame times 32 and 33 ns, respectively; all six displayed comparators remain below their thresholds throughout the shown interval (panel C).

For the transition from the 31 ns starting state, panel D provides conditional-MD evidence using the 20 independent 1 ns branches described in Section 4.4. The constructed target has a descriptor-based discrepancy of 1.32 Å and a descriptive upper-tail rank of $1/21$, whereas the original 32 ns target has a discrepancy of 0.58 Å and rank $21/21$. Thus, the constructed endpoint receives tail support while the native endpoint control is not rejected. This case illustrates a temporally misplaced conformation that escapes the displayed conventional diagnostics but is flagged by ProTAL and supported by conditional endpoint simulations. The evidence concerns endpoint compatibility, not absolute dynamical impossibility.

C.14 FAMILY-LEVEL DETECTION AND ALARM STATISTICS

Figure 3 shows that ProTAL detects temporal inconsistencies across all five corruption families. Its W3 recall is 31.6% for frame swap, 41.4% for block swap, 35.0% for distant-frame replacement, 74.8% for replica splice, and 92.1% for matched splice, yielding a macro-average of 55.0%. The strongest performance occurs for the two splicing families, including boundary-matched splices for which displacement magnitude alone is an incomplete diagnostic. Frame swaps and distant-frame replacements remain more challenging.

To assess whether the comparison depends on the historical candidate-pool difference, we also examine the subsequent evaluation on identical windows. On identical held-out windows, ProTAL

achieves 55.00% macro W3 recall, compared with 12.62% for Transition innovation. The other local dynamic controls obtain 6.04–7.98%, Bond MAE obtains 1.98%, and the three tie-safe static fraction controls detect no labeled edges. The localization advantage therefore persists in the paired evaluation.

Figure 6B complements this case study with an exact-edge false-alarm audit on common inputs. ProTAL's FDP is 32.0%, 27.1%, 50.9%, 35.8%, and 66.0% for Frame Swap, Block Swap, Distant Frame, Replica Splice, and Matched Splice, respectively, lower than every displayed baseline within each family. On the mixture containing the clean windows once and each of the five generated corruption sets once, ProTAL achieves 42.96% exact-edge recall with an FDP of 46.19%; the five dynamic comparators achieve 3.24–9.64% recall with FDPs of 76.45–88.18%. Its FPR on unmodified clean windows is 0.98% (95% protein-bootstrap interval: 0.79–1.18%). Together, these results show higher detection sensitivity and a smaller proportion of label-inconsistent alarms while retaining a low clean-edge alarm rate. They do not imply the lowest FPR for every comparator, and the reported FDPs refer to constructed labels, not universal physical ground truth.

At the 1% empirical FPR budget, ProTAL attains an actual clean FPR of 0.999% and recalls of 25.41%, 40.34%, 26.69%, 75.14%, and 90.06% for frame swap, block swap, distant-frame replacement, replica splice, and matched splice, respectively. Its macro-average recall is 51.53%, compared with 7.24% for the strongest comparator by macro recall, reference-assisted Transition innovation, at an actual FPR of 0.996%. This amounts to a 44.29-percentage-point gain under nearly identical empirical false-alarm rates. ProTAL exceeds every comparator in each corruption family at this budget, with the largest recall values on the two splicing families.

The advantage also holds in every family at all sampled positive FPR budgets from 0.05% to 10%, indicating that the result is not confined to one threshold. Thus, ProTAL provides a favorable sensitivity–false-alarm tradeoff across the evaluated corruption types, while frame swaps and distant-frame replacements remain comparatively difficult. These scans characterize score discrimination on the observed held-out data; they are distinct from the frozen calibration-threshold evaluation in Section 4.4 and do not establish an FPR guarantee on future trajectories.

C.15 Family-wise MD-Reference Analysis of Exclusive Alarms

We expand the five-family summary in the main text to examine where ProTAL's exclusive alarms are enriched for deviations from an empirical MD reference. Figure 8 compares ProTAL with five continuous, reference-free diagnostics separately for each corruption family. We retain the frozen detector thresholds, use exact-edge alarms (W1), and deduplicate physical transitions within each family. The analysis includes all edges with valid paired scores, rather than only the synthetically perturbed edges.

For a baseline B, let A_P and A_B be the alarm sets of ProTAL and B, and let $D \subseteq J$ denote the reference-deviant and adjudicable transition sets. We report

$$\Delta_B = 100 \left[\frac{|(A_P \setminus A_B) \cap D|}{|(A_P \setminus A_B) \cap J|} - \frac{|(A_B \setminus A_P) \cap D|}{|(A_B \setminus A_P) \cap J|} \right], \tag{43}$$

in percentage points (pp). Thus, positive Δ_B indicates a higher reference-deviation rate among ProTAL-only alarms than among baseline-only alarms. Reference outcomes follow the same replica-exclusion protocol as in the main text: every replica contributing frames to the candidate is excluded, and a deviation requires initial-state coverage and an empirical upper-tail rank of at most 0.05 for the transition in both reference folds, together with passing native-control checks. Uncertain outcomes are excluded from J; a zero denominator leaves the comparison undefined.

To account for dependence within proteins, we compute percentile confidence intervals from 2,000 shared bootstrap resamples of the 19 protein system. A resample contributes only when both exclusive subsets have nonzero adjudicable denominators, yielding 1,981–2,000 usable resamples per comparison. The intervals are nominal 95% intervals without multiple-comparison adjustment. Baseline-only transitions are contributed by as few as four systems in some comparisons; large transition counts therefore do not imply many independent observations.

All 20 point estimates across frame swap, block swap, distant-frame replacement, and replica splice favor ProTAL. For replica splice, the differences range from 6.3 to 9.8 pp across the five baselines.

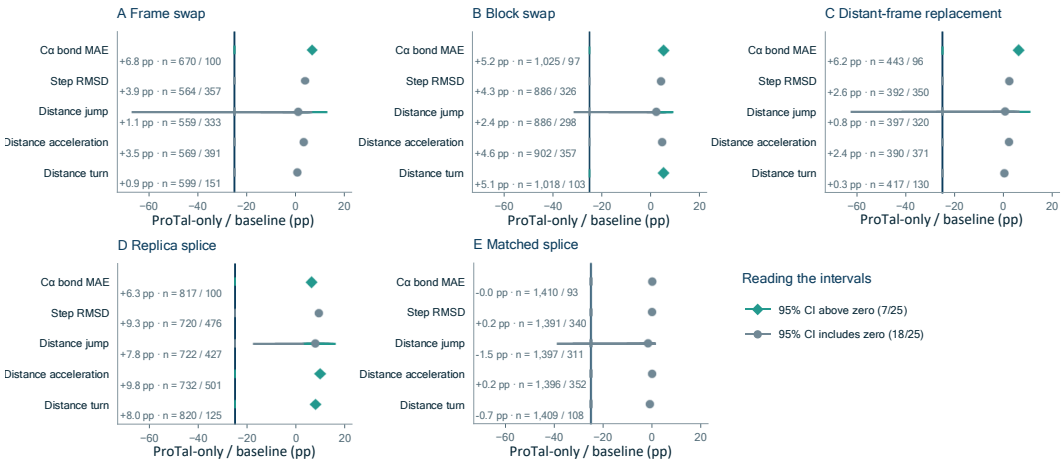


Figure 8: **MD-reference deviation among exclusive alarms, by corruption family.** Panels show **(A)** frame swap, **(B)** block swap, **(C)** distant-frame replacement, **(D)** replica splice, and **(E)** matched splice. Points show Δ_B in Eq. equation 43; horizontal bars show nominal 95% protein-system bootstrap confidence intervals. Teal diamonds identify intervals entirely above zero (7/25), and gray circles identify intervals containing zero (18/25). Each n gives the adjudicable transition counts for ProTAL-only / baseline-only alarms, not independent system counts. Three discrete static controls are omitted because their strict thresholds flag every edge in this analysis, leaving the ProTAL-only denominator empty. Transition innovation is excluded from this reference-free comparison because it shares MD-reference information.

Seven of the 25 intervals lie entirely above zero: the comparisons with Cα bond MAE in all four of these families, Distance turn in block swap and replica splice, and Distance acceleration in replica splice. In contrast, matched-splice differences range from -1.5 to $+0.2$ pp, and all five intervals include zero. The enrichment therefore varies with corruption family, and the intervals support a positive difference in a subset of comparisons.

These results complement the main-text case-coverage analysis by assessing the composition of exclusive alarm sets. Since those sets depend on the comparator and the methods have different realized false-positive rates, Δ_B is neither an accuracy difference nor a comparison at matched FPR. Likewise, deviation from a finite MD reference is evidence of reference incompatibility; non-rejection does not establish physical validity, and rejection does not establish physical impossibility.

C.16 Additional Cases with Conditional-MD Support

We present all twelve selected ProTAL-detected cases supported by the conditional-MD analysis, ordered as five exclusive detections followed by seven shared detections. In the first five figures (U01–U05; Figures 18–22), ProTAL flags the selected exact edge (W1), whereas none of the five continuous diagnostics or Transition innovation (TI) does. The three discrete static controls also remain silent under the tie-safe protocol. The remaining seven figures (S01–S07; Figures 23–29) show cases detected by ProTAL and at least one comparator. Exclusivity refers to the selected edge under the frozen thresholds; neighboring edges may still be detected by a baseline. These cases span eight proteins and share some MD starting states, so they are not twelve independent performance evaluations.

Each figure connects the constructed transition to its detector response and conditional-MD evidence. **Panel A** shows four saved 64-site Cα conformations, rigidly aligned to the selected constructed left frame and displayed in a common orthographic view. Orange traces denote the constructed window, blue traces show the unmodified window at altered positions, and dark traces are unchanged. Top labels give nominal window times, while replacement annotations specify actual source replicas and times. **Panel B** shows scores divided by each method's frozen threshold. The horizontal line at one marks the alarm boundary, the vertical line identifies the selected edge, and the short ticks below the curves mark constructed error edges. The gray curves correspond to Cα bond

MAE, Step RMSD, Distance jump, Distance acceleration, and Distance turn; TI is shown separately. Adjacent-edge responses are retained.

Panel C compares the constructed target and the native 1 ns successor with 20 saved independent 1 ns conditional-MD endpoints from the actual starting conformation. Each target forms a separate 21-member pool with those endpoints. For the 1,830 nonlocal Cα distances with sequence separation at least four, let z_i denote the distance descriptor of pool member i. We compute

$$d_{ij} = \sqrt{\frac{1}{1830}\|z_i - z_j\|_2^2}, \qquad a_i = \frac{1}{20}\sum_{j \neq i} d_{ij}, \qquad r = \frac{\#\{i : a_i \geq a_{\text{target}}\}}{21}. \tag{44}$$

Gray circles show MD endpoints and colored diamonds show the targets; discrepancies are recomputed separately in the two pools. The ranks include the target and are descriptive, rather than calibrated p-values. For S01, S04, S05, and S07, the selected left frame is itself replaced: the native control in C follows its actual source conformation and therefore differs from the blue unmodified window in A.

Across all twelve cases, the constructed target has rank $1/21$, whereas native-control ranks range from $2/21$ to $21/21$. Together with the score traces, the first five cases illustrate transitions missed by the displayed comparators at the selected edge despite receiving conditional-MD support. The seven shared cases show agreement between ProTAL and conventional diagnostics. This evidence concerns compatibility of the tested endpoints with sampled continuations, without establishing that an entire trajectory is physically impossible.

C.17 Additional Analyses of Detection Utility

We supplement the main-text low-FPR analysis with four assessments of practical detection utility: prioritizing a limited inspection budget, variation across proteins, temporal localization, and complementarity with existing diagnostics. These analyses use the saved scores on 19 held-out proteins and common valid positions across methods; ProTAL is labeled V12 in the original source figures. Each clean or corrupted cohort contributes 38,340 valid window–edge occurrences. Overlapping windows may repeat a physical transition, so these counts describe inspection positions rather than independent events. Frozen thresholds are retained for all analyses except budget-based ranking.

Prioritizing limited inspection budgets. Figure 9 evaluates ranking on the mixture of clean data and all five corruption families, comprising 230,040 positions and 11,898 labeled positives. At a requested inspection budget of 1%, ProTAL selects 2,300 positions, including 2,217 positives, for 96.39% precision and 18.63% pooled recall. Under the same requested budget, the strongest comparator, reference-assisted Transition innovation (TI), achieves 29.32% precision and 5.65% recall. At 5%, ProTAL reaches 47.08% recall with 48.70% precision, compared with TI's 16.77% and 17.34%, respectively. These results show that high ProTAL scores concentrate labeled errors near the top of the inspection list. The budget counts window–edge occurrences; it does not measure human review time.

Variation across held-out proteins. Figure 10 tests whether pooled performance conceals concentration on a few proteins. Averaging the five family recalls within each protein and then weighting proteins equally gives ProTAL a mean recall of 53.34%, compared with 13.02% for the strongest continuous baseline, Step RMSD. ProTAL exceeds Step RMSD on 17 of 19 proteins and each of the other four continuous diagnostics and TI on all 19. Performance remains heterogeneous: ProTAL's median recall is 47.99%, with a range of 30.23–89.45%; its clean FPR ranges from 0.25% to 1.94%. The paired mean recall gain over Step RMSD is 40.32 percentage points (nominal 95% interval: 32.43–47.78), using 1,000 paired protein-level bootstrap resamples. These intervals have no multiple-comparison adjustment and condition on the trained model and frozen thresholds. Paired clean-FPR difference intervals include zero for all six comparators, so the evidence for consistent recall gains is stronger than that for a uniform FPR ordering.

Temporal localization and its false-alarm cost. Figure 11 varies the allowed displacement from a labeled error edge. ProTAL achieves 51.36% equal-family recall at radius zero (W1); allowing one neighboring edge on either side (W3) gives 55.18%. The corresponding clean-neighborhood

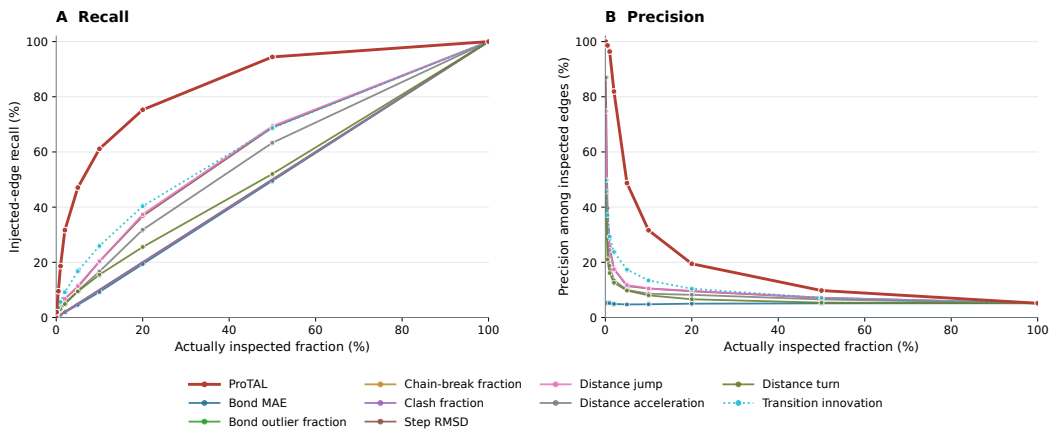


Figure 9: **Detection under limited inspection budgets.** Pooled injected-error recall (left) and precision among inspected positions (right) versus the actually inspected fraction of the clean-plus-corruption mixture. Each method ranks its saved scores and includes only whole tied-score blocks that fit within the requested budget, without label-based tie breaking. Markers are evaluated budgets; lines are visual guides. Precision is undefined when no position is inspected. All ten methods are retained; TI uses reference information.

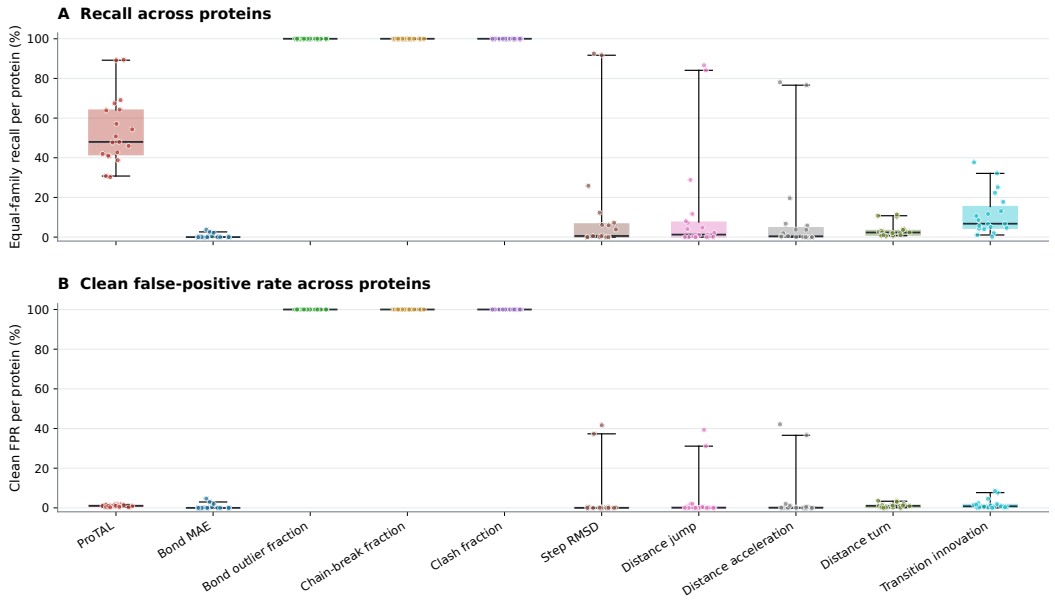


Figure 10: **Recall and clean FPR across 19 held-out proteins.** Each point is one protein. Recall first averages the five corruption families within that protein. Boxes show medians and interquartile ranges; whiskers follow the 5th–95th-percentile rule, and all finite protein-level observations are displayed. These distributions are not confidence intervals or repeated training runs. Under the primary strict thresholds, the three discrete static fractions alarm everywhere, producing both 100% recall and 100% FPR; their recall therefore does not indicate discrimination.

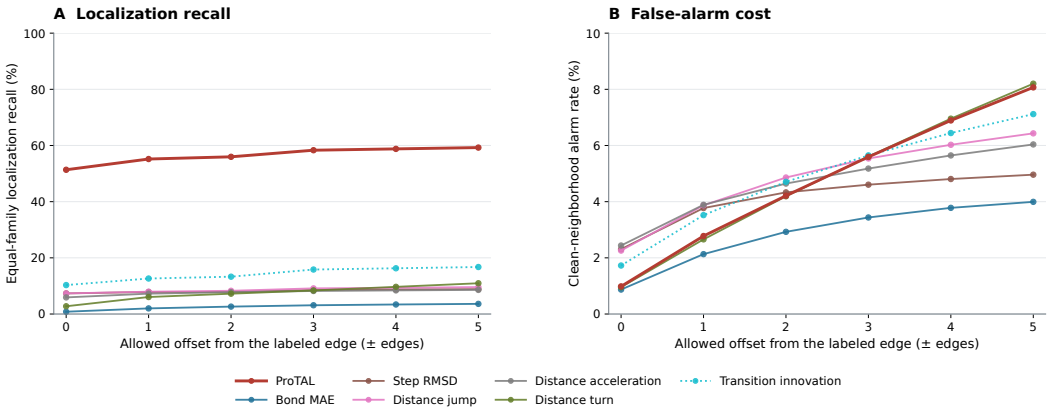


Figure 11: **Localization tolerance and clean-neighborhood alarms.** Left: equal-family recall when an alarm within $\pm r$ edges counts as a hit. Right: the fraction of eligible clean centers with at least one alarm in the same neighborhood, shown on a 0–10% scale. Radius zero is W1 and radius one is W3. Neighborhoods are truncated at window boundaries, and every included score must be finite. One alarm may cover multiple labeled centers, so these curves are not one-to-one event matching. For readability, the three degenerate static-fraction controls (bond outlier, chain-break, and clash fractions) are omitted from the plotted curves: their strict zero thresholds flag every valid position, yielding 100% recall and 100% clean-neighborhood alarm rates at every radius. All other methods and sampled radii are retained.

alarm rate increases from 0.98% to 2.78%. At radius five, recall reaches 59.26%, while the clean-neighborhood alarm rate rises to 8.07%. Thus, most of the measured recall is already obtained at the exact edge, and wider tolerances yield modest gains with increasing false-alarm exposure. Neighborhoods are clipped at window boundaries, and a center is retained only when all scores in its clipped neighborhood are finite. Eligible centers therefore vary with radius; this mask differs from historical W3 summaries that allowed any available alarming neighbor.

Additional detections beyond conventional diagnostics. Figure 12 measures complementary alarms on the five constructed cohorts, with 11,898 positive and 179,802 negative positions. Adding ProTAL to the fixed union of the five continuous reference-free diagnostics finds 4,059 additional positives, increasing pooled recall from 11.42% to 45.54%. It also adds 3,645 negative alarms, raising FPR on the constructed negative positions from 5.11% to 7.14%. Conversely, adding that diagnostic union to ProTAL contributes 307 positives and 8,819 negative alarms. ProTAL also contributes 4,131 positives beyond TI, accompanied by 3,905 negative alarms. These results quantify substantial complementary coverage together with its cost. They use constructed labels and pooled counts, separately from the main-text MD-supported case analysis and equal-family recall.

D Extended Related Work

The main-text discussion distinguishes conformational sampling, dynamical modeling, and transition localization. Here we develop these connections in more detail, emphasizing the quantity each approach learns, the information available to it, and the evidence required to interpret its output. Table 10 summarizes these distinctions; it is a comparison of tasks rather than a performance ranking.

D.1 Conformational Sampling and the Meaning of Generated Order

An equilibrium ensemble describes the populations of molecular configurations. Boltzmann generators approach this problem through invertible transformations between a simple latent distribution and molecular coordinates, with statistical-mechanical training and reweighting Noé et al. (2019). More recent protein samplers exploit learned structural priors. Str2Str perturbs an input conformation and reverses the perturbation with a score model, enabling sampling around a structure without target-specific MD training Lu et al. (2024). AlphaFlow adapts AlphaFold and ESMFold using flow

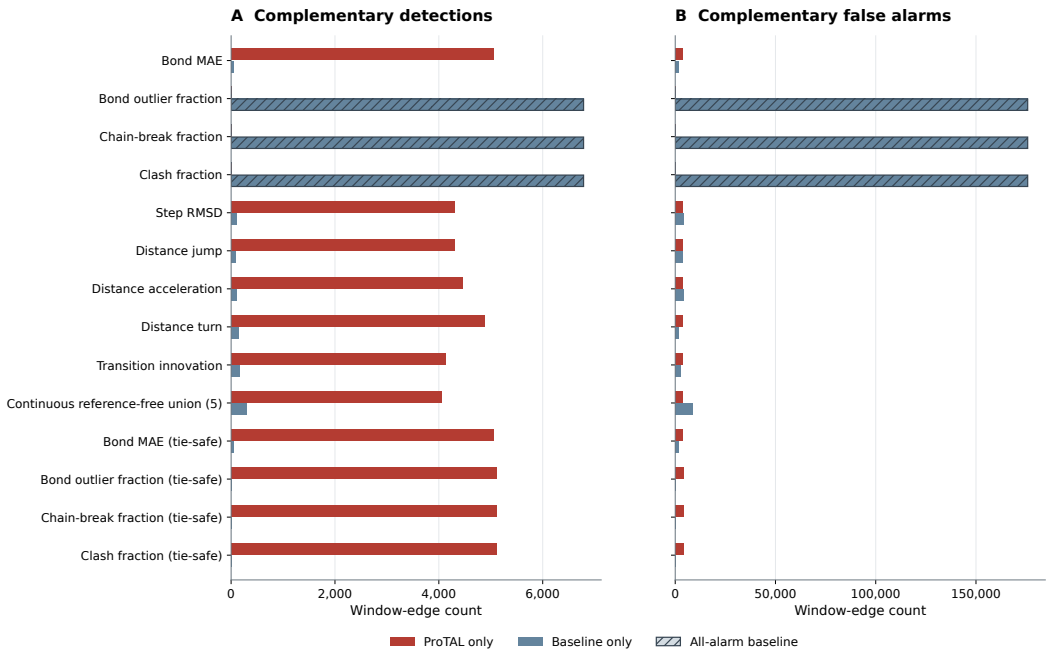


Figure 12: **Pairwise complementary detections and false alarms.** Each row compares ProTAL with one method or the predefined union of five continuous reference-free diagnostics. Red bars count ProTAL-only alarms and blue bars comparator-only alarms, separately on constructed positives (left) and negatives (right). Red counts are the additions obtained by adding ProTAL to that comparator; blue counts describe the reverse addition. The populations contain 11,898 positives and 179,802 negatives, with no additional clean cohort. Hatched bars identify strict all-alarm baselines; tie-safe sensitivity rows are also retained. Counts concern window–edge occurrences and injected labels, rather than globally unique events or MD-supported cases.

matching to generate conformational ensembles Jing et al. (2024a). These methods make different architectural and training choices, but all motivate evaluating a collection of plausible structures beyond a single predicted fold.

Structure language modeling changes the representation used for this generative task. Lu et al. (2025) encode conformations into discrete structural tokens and model their sequence-conditioned distribution; ESMDiff implements this idea with masked diffusion. The order of token generation or denoising is an algorithmic order, which should not be identified with molecular time. BioEmu extends ensemble modeling by combining MD, structural data, and experimental protein stability measurements, and evaluates conformational coverage and thermodynamic properties Lewis et al. (2025). Its statistically independent samples are intended to represent equilibrium populations rather than consecutive physical states.

This distinction is central to our problem formulation. A sampler can successfully reproduce a conformational distribution without specifying how the sampled states should be connected at a given time interval. Reordering the same samples leaves any permutation-invariant ensemble statistic unchanged. It follows that temporal compatibility is an additional question only when a sequence is assigned a physical interpretation. ProTAL is designed for that setting. Applying its scores to an arbitrarily ordered equilibrium sample collection would conflate the absence of a prescribed trajectory with an error in the ensemble sampler. Our benchmark instead preserves MD-sampled constituent structures while deliberately changing their placement in observed trajectories.

D.2 Transition Models, Trajectory Generators, and Sampling Objectives

Methods trained on time-separated observations address information absent from a purely equilibrium model. Timewarp learns a conditional normalizing-flow proposal from MD and combines it with a Metropolis–Hastings correction to target the Boltzmann distribution Klein et al. (2023). The learned proposal exploits time-coarsened transitions, while the corrected chain serves an equilibrium sampling objective. These roles should be distinguished: a chain that has the desired stationary distribution does not thereby reproduce physical kinetics at every sampling step. Timewarp's reported transfer to unseen small peptides concerns a different output and scale from our transition-localization benchmark.

Implicit Transfer Operator (ITO) learning models conditional evolution across multiple lag times using equivariant diffusion, including coarse-grained representations of partially observed systems Schreiner et al. (2023). Its evaluation considers whether direct large-lag predictions agree with repeated shorter-lag evolution. BoPITO adds Boltzmann priors to improve data efficiency and equilibrium behavior Viguera Diez et al. (2025). These works illustrate that temporal consistency and long-time populations can be addressed jointly, although they remain distinct properties. ProTAL's multiple prediction horizons serve another purpose: discrepancies from overlapping forecasts supply evidence for locating a suspicious observed edge, rather than generating a transition kernel.

MDGen takes a trajectory-level approach, conditioning on selected frames or molecular components to support forward simulation, interpolation, upsampling, and inpainting Jing et al. (2024b). Its tetrapeptide experiments assess dynamical observables as well as structural distributions; the protein experiments explore ensemble modeling at a larger molecular scale. Thus, our motivation is not that trajectory generators ignore temporal evaluation. The complementary question is whether a diagnostic can identify where a particular candidate sequence conflicts with learned temporal context. We evaluate this question using controlled ATLAS perturbations and trajectories generated by a cadence-adapted MDGen model (Section 4.3).

D.3 Structural Validation and Statistical Kinetics

Structural validation is already capable of detailed local diagnosis. MolProbity, for example, combines all-atom contact analysis with covalent-geometry and torsion-angle criteria Chen et al. (2010). Such criteria identify defects within a conformation. They address a different question from whether an otherwise plausible conformation belongs between its observed temporal neighbors. Our Cα bond and clash controls are deliberately restricted structural diagnostics; they are not implementations of the full MolProbity suite, and the comparisons should not be read as a benchmark against all-atom validation.

Kinetic analysis explicitly uses temporal information. Markov-state models estimate transitions between discretized configurations, with validation procedures assessing whether the resulting model reproduces the observed kinetics Prinz et al. (2011). VAMPnets learn a mapping from molecular coordinates to kinetic states through a variational objective Mardt et al. (2018). Time-lagged autoencoders learn nonlinear low-dimensional coordinates by reconstructing future features Wehmeyer & Noé (2018). These approaches organize molecular dynamics into states, slow variables, and transition statistics, providing information that framewise structural checks cannot supply.

Their relationship to anomaly localization is complementary rather than exclusive. A kinetic model can supply a transition likelihood or a residual that could be adapted into a local diagnostic. Doing so requires choices about molecular features, state discretization, lag time, reference coverage, and the conversion of statistical evidence into an alarm. ProTAL directly learns edge scores across proteins and uses a fixed clean-calibration operating point. It does not estimate equilibrium populations, rate constants, or validated metastable states. Conversely, a low ProTAL score cannot establish those global properties. Our reference-assisted Transition innovation baseline and the separate MD adjudication procedures provide specific comparisons with reference information; they do not exhaust the possible kinetic-model-based detectors.

D.4 Geometric Representations and Predictive Surprise

The representation must distinguish meaningful molecular motion from changes in coordinate convention. GVP networks provide a protein-structure architecture that processes scalar and vector features together Jing et al. (2021). ProTAL uses a GVP-style encoder to obtain pooled frame representations and supplements them with residue-pair changes. Pair distances remove sensitivity to global rigid alignment, but cannot by themselves distinguish mirror images; chirality-sensitive information is therefore retained in the predictive pathway. The two pathways operate at different levels of aggregation: pooled molecular states support temporal forecasting, while residue tokens retain the distribution of pairwise changes before temporal integration. This is a design rationale; the relative contributions of the pathways require ablation evidence.

Learning through prediction has precedents beyond molecular dynamics. Contrastive Predictive Coding predicts future latent information using an autoregressive context and a contrastive objective Oord et al. (2018). I-JEPA predicts the embeddings of masked image regions from visible context Assran et al. (2023), and V-JEPA learns video representations through feature prediction Bardes et al. (2024). These approaches motivate learning useful states without reproducing every detail of an observation. They do not imply that a representation trained for prediction automatically yields a calibrated anomaly detector.

LeWorldModel makes the connection to physical surprise explicit: it learns latent dynamics from pixels using prediction and Gaussian-distribution regularization, and evaluates surprising events in visual environments Maes et al. (2026). Our use of latent prediction is conceptually related, but the optimization and supervision differ. LeWorldModel uses SIGReg and avoids stop-gradient and moving-average target encoders; ProTAL uses detached prediction targets and residual baselines, additional clean representation objectives, and explicit corruption labels for detection. It also combines independent forward and backward predictors with a separate residue-geometry pathway. We therefore do not inherit LeWorldModel's training guarantees or treat its visual surprise results as evidence for molecular validity.

For stochastic molecular motion, even an accurate predictive representation can assign a large error to a legitimate outcome. Partial Cα observations also omit velocities, solvent, and other variables relevant to evolution. In ProTAL, prediction error is consequently an input to a learned contextual decision rather than a physical verdict. Bidirectional evidence is appropriate to retrospective inspection of a complete window, but it uses information unavailable to an online detector. Any future comparison with causal forecasting systems should explicitly match the observations available at the time an alarm is produced.

D.5 Contextual Time-series Anomaly Detection and Synthetic Supervision

Time-series anomaly detectors provide several mechanisms for turning temporal context into an anomaly score. Anomaly Transformer contrasts a locality-based prior association with a data-

dependent series association, using a minimax objective to amplify their discrepancy Xu et al. (2022). DCdetector instead learns complementary patch-wise and in-patch representations through a dual-attention contrastive construction Yang et al. (2023). Their common relevance is that an observation can be assessed through its relationships with the surrounding series, rather than through its isolated magnitude. Neither result establishes which representation or scoring rule will work best for protein trajectories.

DADA studies generalization across time-series domains through extensive pretraining, adaptive information bottlenecks, and dual adversarial decoders Shentu et al. (2025). Its training includes injected abnormal patterns, and inference uses the variability of reconstructions under complementary masks as an anomaly score. This provides a useful comparison for our use of constructed corruptions: artificial anomalies can shape a decision boundary even when naturally annotated failures are unavailable. The constructions differ, however. ProTAL reorders or replaces MD-sampled frames and joins trajectory segments, so the intended distinction concerns temporal placement while preserving plausible constituent conformations.

This supervision should be described precisely. The clean prediction component learns from observed dynamics without anomaly annotations, but the detection heads and their later interaction with the encoder use explicit labels. The overall procedure is therefore supervised on synthetic temporal anomalies. Those labels define the evaluated task; they do not exhaust every possible failure of a generated or simulated trajectory. A model could also exploit artifacts specific to a corruption procedure. Protein-separated evaluation, matched-boundary splices, and selected MD checks probe this concern. The time-warp and MDGen evaluations further assess transfer beyond the training corruptions (Section 4.3), although low time-warp recall and limited generated-trajectory coverage leave broader generalization unresolved.

Section 4.3 evaluates adaptations of time-series detectors and a kinetic model to protein trajectories. These results concern the evaluated molecular inputs, supervision regimes, and adjacent-frame scoring protocols, rather than the methods' original benchmarks.

D.6 Evaluation, Reference Information, and Interpretation

The evaluation unit can change the apparent difficulty of temporal anomaly detection. Kim et al. (2022) show that point adjustment, which credits an entire anomalous segment after a detection inside it, can substantially inflate reported performance. Our W3 metric uses a fixed one-edge tolerance around each labeled transition; it does not fill an arbitrarily long segment. Nevertheless, neighboring labels can share a nearby alarm, so W3 should be interpreted as tolerant localization rather than exact timing. Reporting W1 alongside W3 makes that distinction visible, while the family-level results expose variation across corruption mechanisms.

Alarm calibration is a separate issue from score discrimination. The frozen threshold is estimated on clean calibration proteins and evaluated on held-out proteins. The low-FPR threshold scans instead describe trade-offs obtainable from held-out scores; they are not independent estimates of a deployment threshold's guarantee. Likewise, clean FPR measures alarms on unmodified observations, whereas FDP depends on the prevalence and composition of anomalies in the evaluated mixture. A low value of one does not determine the other. These distinctions are necessary when relating our results to papers that report optimized F1 scores, range-based metrics, or different anomaly mixtures.

Finally, the information used to produce an alarm should be separated from information used to investigate it. ATLAS provides standardized protein MD data Vander Meersche et al. (2024), from which we construct protein-disjoint subsets. ProTAL requires only the candidate window and its metadata at inference. The clean partner is used during training, while independent replicas and new conditional-MD runs support selected post hoc analyses. Those analyses offer evidence about endpoint compatibility under specified references and simulation conditions. They do not assign a full path probability or turn constructed labels into universal physical ground truth. This separation defines ProTAL's present contribution: localized, calibrated anomaly evidence that can guide further inspection alongside structural and kinetic assessment.

Table 10: **Scope of related approaches.** The rows summarize the objectives discussed in Appendix D. They do not imply that the methods cannot be adapted to other tasks, and do not report an empirical comparison.

Approach	Typical output or target	Relationship to ProTAL
Conformational samplers	Structures, diversity, or equilibrium populations; e.g., Str2Str, AlphaFlow, SLM, BioEmu	Establish properties of an ensemble; its sample order need not define a physical trajectory.
Learned transition and trajectory models	Conditional transitions or complete trajectories; e.g., Timewarp, ITO, MDGen	Model evolution or accelerate sampling; ProTAL scores an observed sequence and does not generate it.
Structural validation	Geometric defects within a conformation; e.g., MolProbity	Complements contextual checks when constituent frames remain structurally plausible.
Kinetic-state models	States, transition operators, and slow dynamics; e.g., MSMs, VAMPnets	Characterize statistical kinetics; adaptation to local alarms requires a scoring and calibration protocol.
Predictive representations	Latents predictive of unobserved content; e.g., CPC, I-JEPA, V-JEPA, LeWM	Motivate predictive evidence; neither latent error nor representation quality alone certifies molecular validity.
Time-series detectors	Anomaly scores from temporal relationships; e.g., Anomaly Transformer, DCdetector, DADA	Closely related detection objectives; molecular inputs and edge-level evaluation require explicit adaptation.
ProTAL	Offline adjacent-frame anomaly scores from predictive and geometric evidence	Supervised on controlled temporal corruptions; no paired reference is needed at inference.

D.7 Additional Ablation Results

We complement the fixed-calibration results in the left block of Table 4 with detection recall by perturbation family and the corresponding false-alarm rates. Figure 13 shows that ProTAL's advantage over Transitions-only, No-forecast, and Single-scale is most pronounced for matched splices. The differences are smaller for frame swaps and replica splices, whereas No-forecast and Transitions-only achieve higher recall than ProTAL on distant-frame perturbations. Thus, the improvement in macro-average recall reflects gains on particular perturbation families rather than uniform improvements across all types.

Motion-only exhibits a different detection profile: it detects many replica splices but few matched splices. Geometry-TCN and Geometry-Transformer also have low matched-splice recall. The full model's stronger performance on this family, compared with either branch alone, is consistent with the complementary contributions of contextual prediction and residue-pair geometry.

Figure 14 reports false positives on clean edges. ProTAL achieves an FPR of 0.915%, below Transitions-only (1.023%), No-forecast (1.013%), and Single-scale (0.996%), and close to Motion-only (0.920%). Geometry-TCN and Geometry-Transformer attain lower FPRs, but also substantially lower recall. At the reported operating points, their reduced false-alarm rates therefore accompany more missed anomalies, highlighting the need to interpret both measures jointly.

D.8 Ablation at Matched False-Positive Rates

We supplement the matched-FPR results in the right block of Table 4 with comparisons across perturbation families and empirical test clean-edge FPR budgets. At the 1% cap, all seven models attain the same empirical FPR of 1% (2,137 false positives among 213,714 clean edges). Figure 15 summarizes their W1 and W3 macro recall, while Figure 16 provides the W3 breakdown across perturbation families. ProTAL's advantage over Transitions-only, No-forecast, and Single-scale is most pronounced for matched splices; the improvement is not uniform across all families.

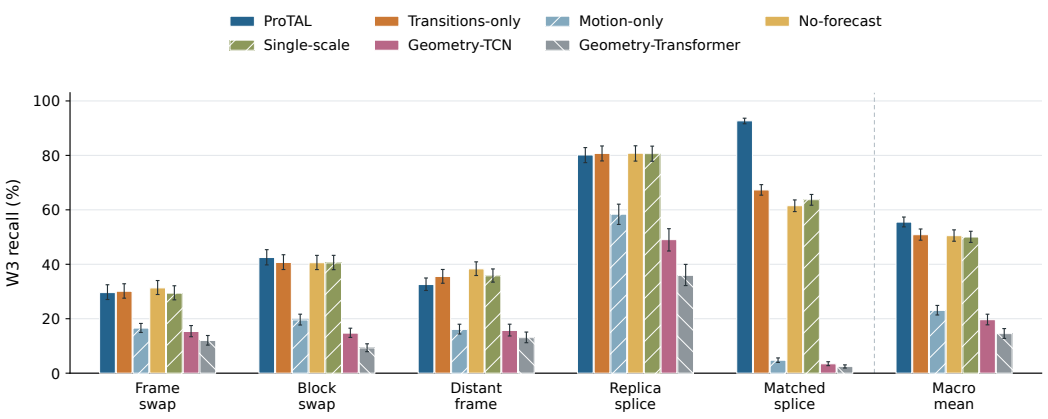


Figure 13: W3 recall by perturbation family on Test82 for ProTAL and its six comparison variants. The five families are frame swap, block swap, distant frame, replica splice, and matched splice. The group to the right of the dashed line reports the equally weighted mean across these families. Higher values indicate better detection.

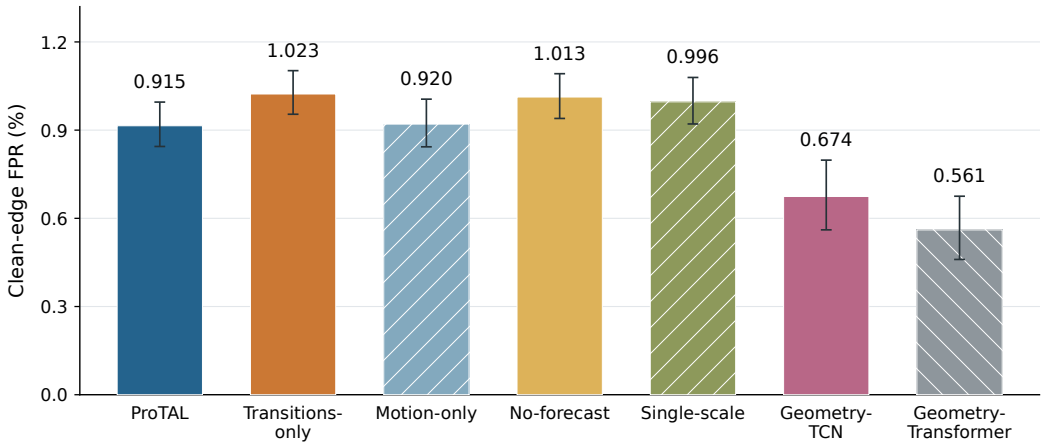


Figure 14: Clean-edge false-positive rates on Test82 for the same seven methods. Labels above the bars report FPR values in percent. Lower values indicate fewer false alarms; detection recall is shown in Figure 13.

Figure 17 extends the comparison across false-alarm budgets. ProTAL achieves the highest W3 macro recall at every evaluated FPR cap from 0.05% to 2%, showing that its advantage persists across the reported operating points. These results describe empirical test-set operating points rather than thresholds fixed through independent calibration.

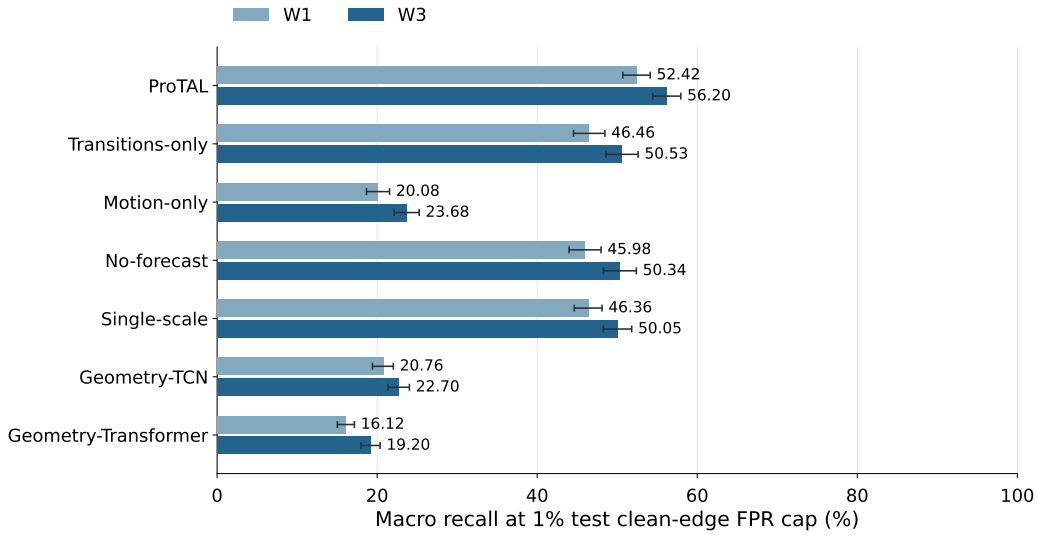


Figure 15: W1 and W3 macro recall on Test82 at a matched empirical clean-edge FPR of 1%. W1 requires exact-edge localization, whereas W3 allows a one-edge tolerance.

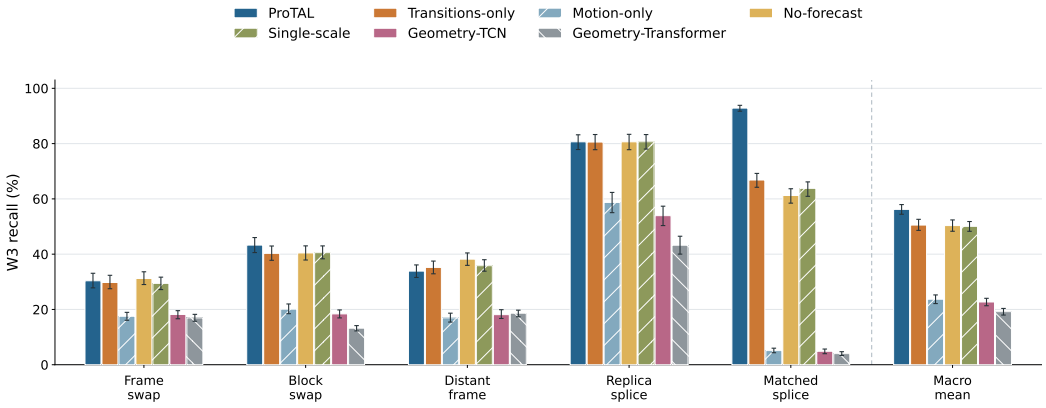


Figure 16: W3 recall by perturbation family at the same 1% empirical clean-edge FPR. The group to the right of the dashed line reports the equally weighted mean across the five families.

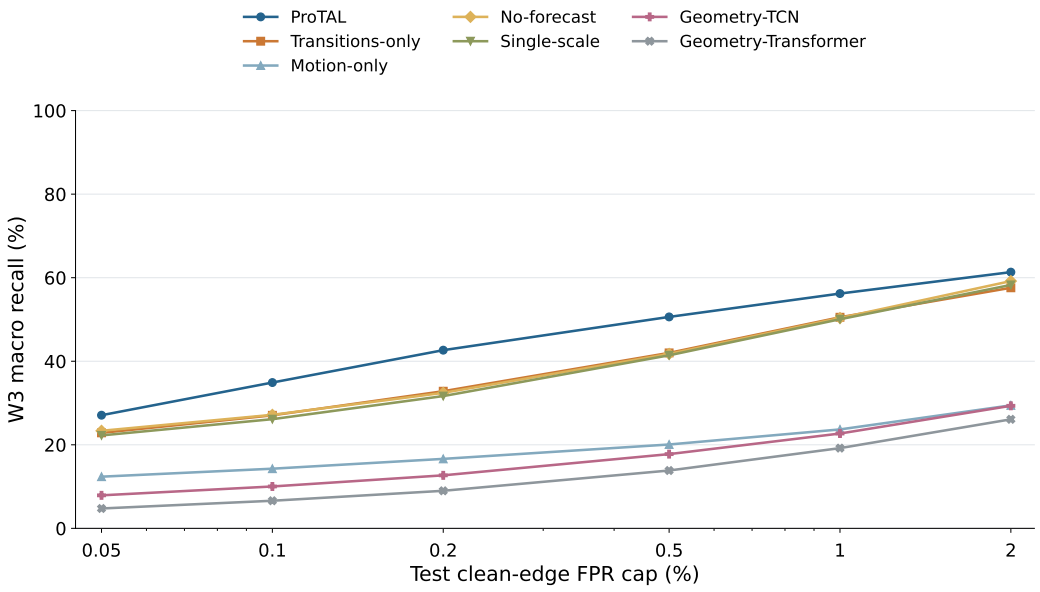


Figure 17: W3 macro recall across empirical test clean-edge FPR caps of 0.05%, 0.1%, 0.2%, 0.5%, 1%, and 2%. ProTAL achieves the highest recall at each evaluated cap; lines connect the evaluated operating points.

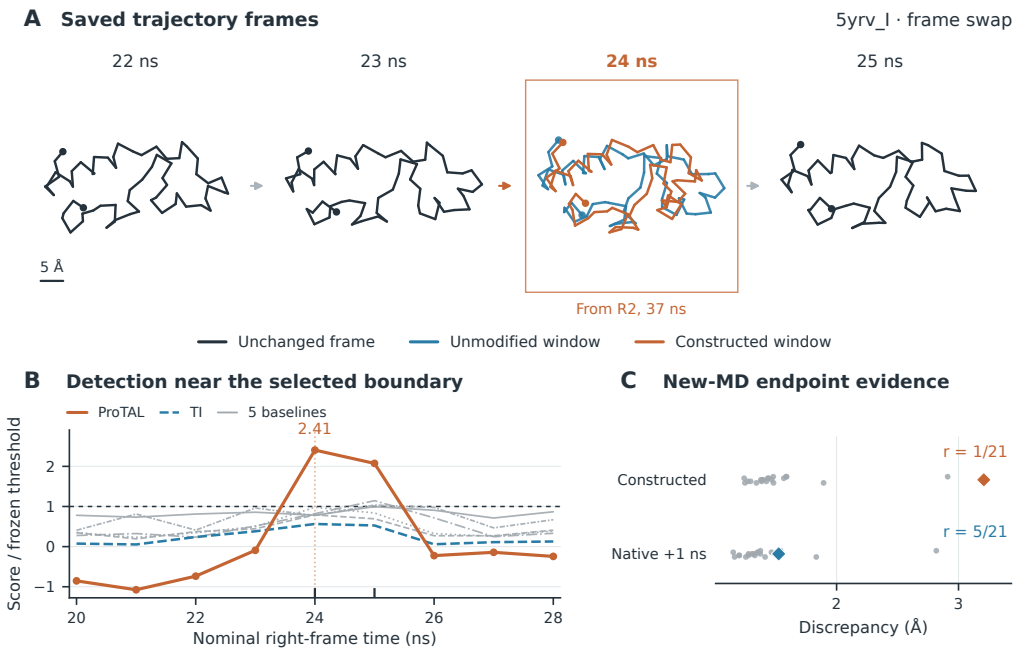


Figure 18: **U01: Unique W1 detection in `5yrv_I` (frame swap). A:** aligned trajectory frames; **B:** normalized detection scores; **C:** conditional-MD endpoint discrepancies. The selected constructed transition joins R2 at 23 ns to R2 at 37 ns; the native control is R2 at 24 ns. The constructed and native targets have descriptive ranks $1/21$ and $5/21$, respectively. Panel definitions and the W1 criterion are given in Section C.16.

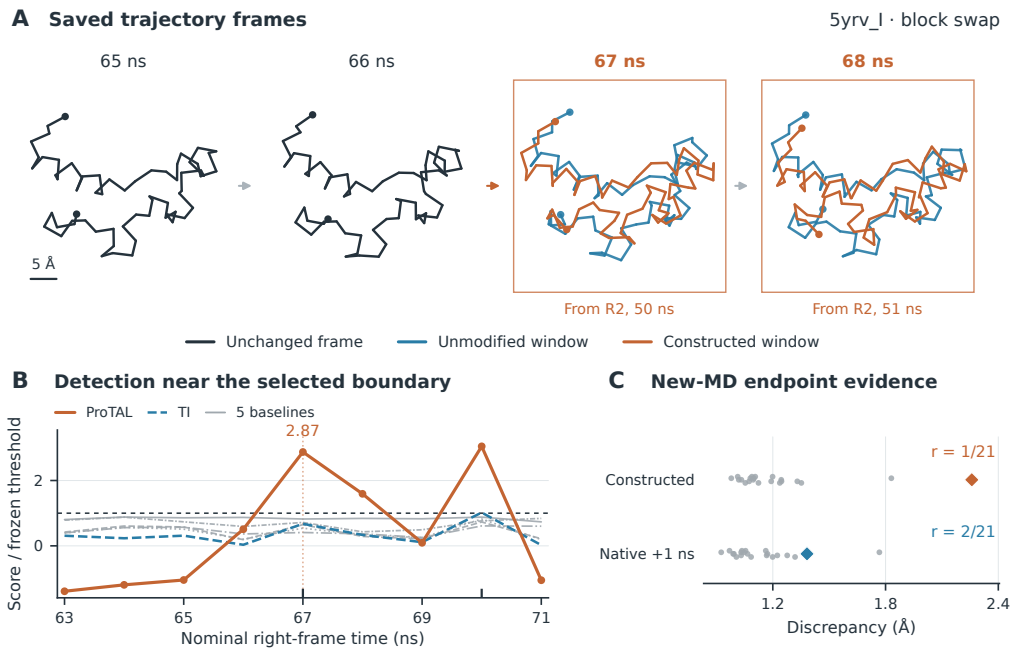


Figure 19: **U02: Unique W1 detection in `5yrv_I` (block swap). A:** aligned trajectory frames; **B:** normalized detection scores; **C:** conditional-MD endpoint discrepancies. The selected constructed transition joins R2 at 66 ns to R2 at 50 ns; the native control is R2 at 67 ns. The constructed and native targets have descriptive ranks $1/21$ and $2/21$, respectively. Panel definitions and the W1 criterion are given in Section C.16.

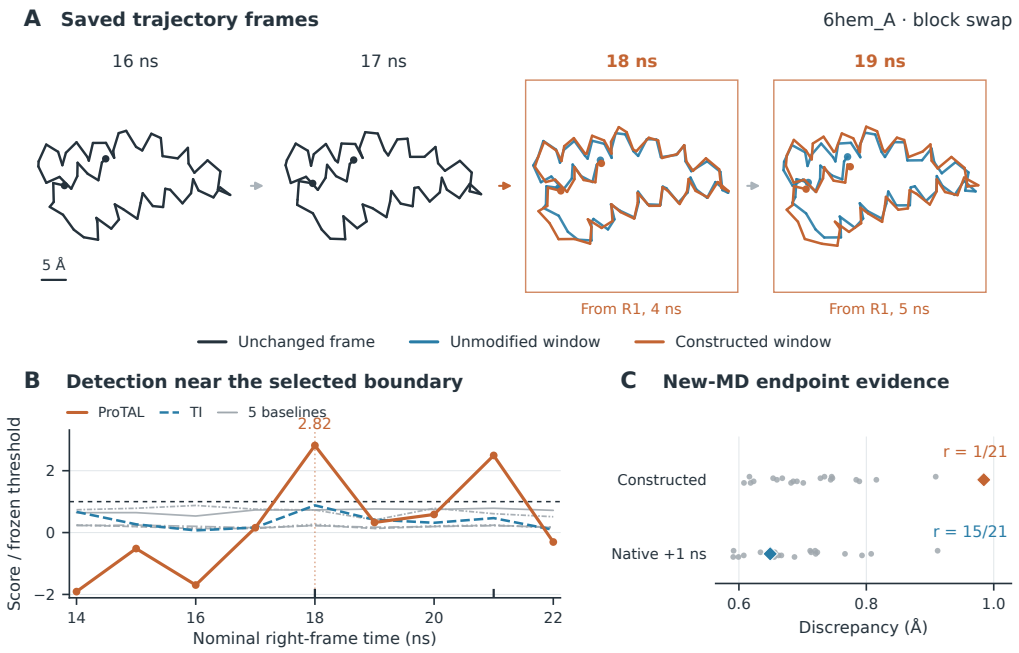


Figure 20: **U03: Unique W1 detection in `6hem_A` (block swap). A:** aligned trajectory frames; **B:** normalized detection scores; **C:** conditional-MD endpoint discrepancies. The selected constructed transition joins R1 at 17 ns to R1 at 4 ns; the native control is R1 at 18 ns. The constructed and native targets have descriptive ranks $1/21$ and $15/21$, respectively. Panel definitions and the W1 criterion are given in Section C.16.

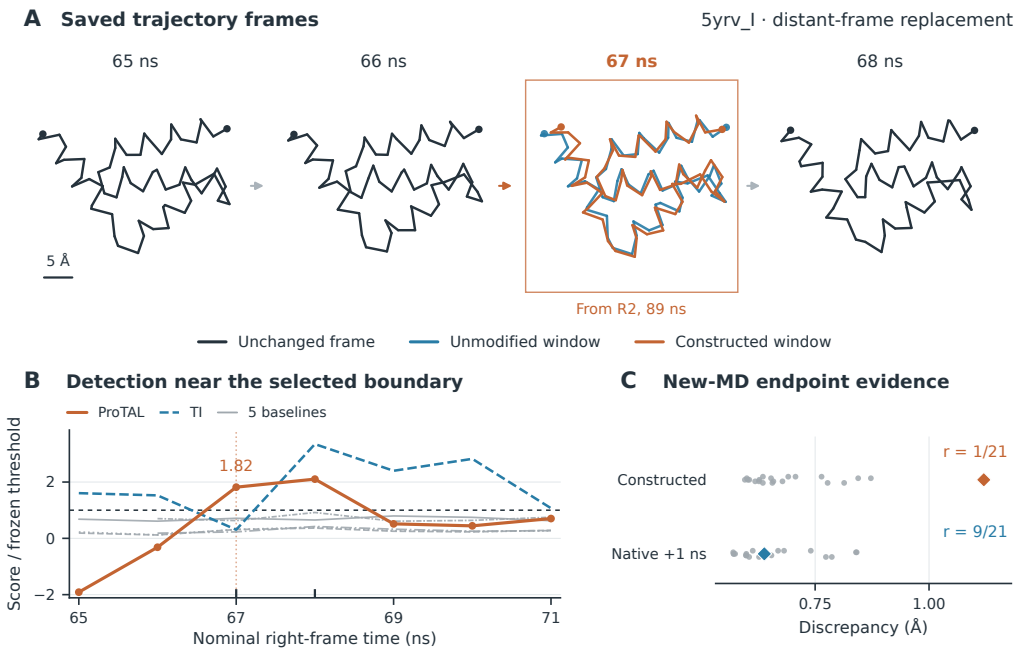


Figure 21: **U04: Unique W1 detection in `5yrv_I` (distant-frame replacement). A:** aligned trajectory frames; **B:** normalized detection scores; **C:** conditional-MD endpoint discrepancies. The selected constructed transition joins R2 at 66 ns to R2 at 89 ns; the native control is R2 at 67 ns. The constructed and native targets have descriptive ranks $1/21$ and $9/21$, respectively. Panel definitions and the W1 criterion are given in Section C.16.

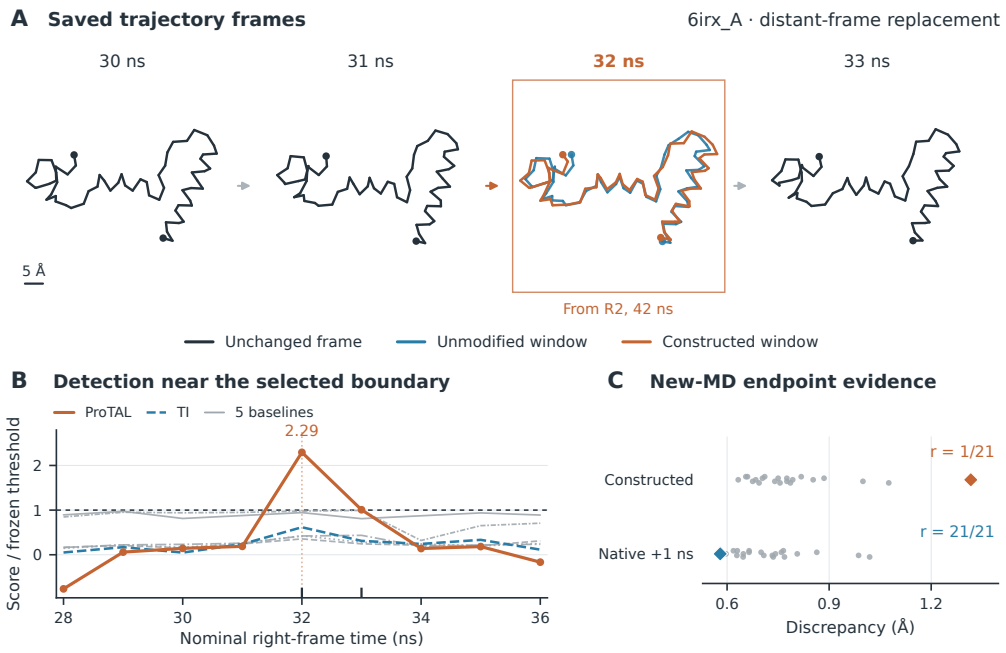


Figure 22: **U05: Unique W1 detection in `6irx_A` (distant-frame replacement).** **A:** aligned trajectory frames; **B:** normalized detection scores; **C:** conditional-MD endpoint discrepancies. The selected constructed transition joins R2 at 31 ns to R2 at 42 ns; the native control is R2 at 32 ns. The constructed and native targets have descriptive ranks $1/21$ and $21/21$, respectively. Panel definitions and the W1 criterion are given in Section C.16.

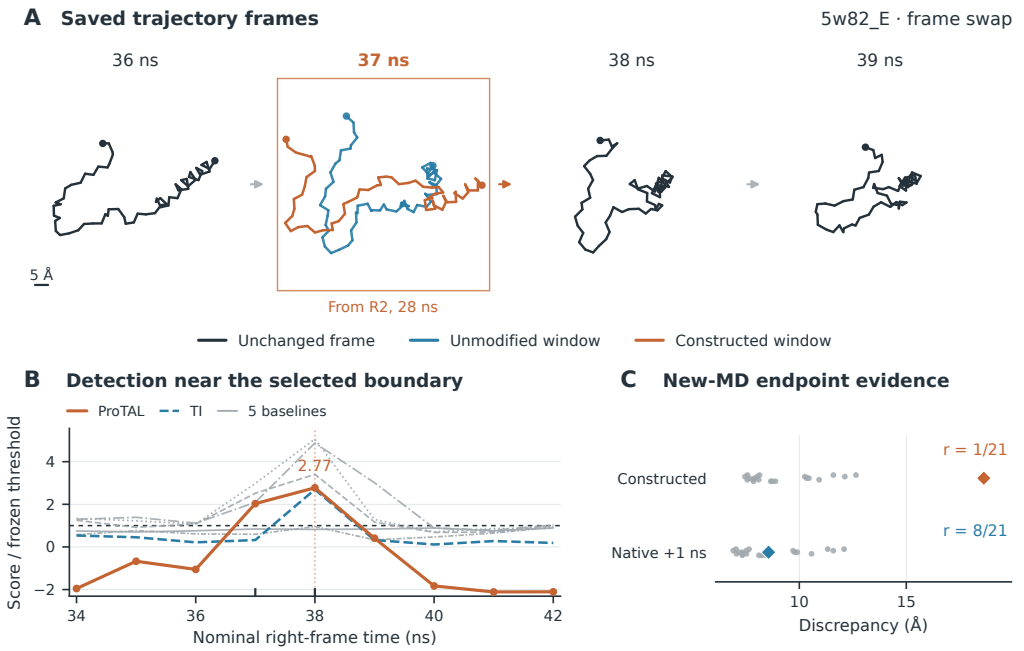


Figure 23: **S01: Shared W1 detection in `5w82_E` (frame swap). A:** aligned trajectory frames; **B:** normalized detection scores; **C:** conditional-MD endpoint discrepancies. The selected constructed transition joins R2 at 28 ns to R2 at 38 ns; the native control is R2 at 29 ns. The constructed and native targets have descriptive ranks $1/21$ and $8/21$, respectively. The native control follows the actual replaced left frame and differs from the blue unmodified window in A. Panel definitions and the W1 criterion are given in Section C.16.

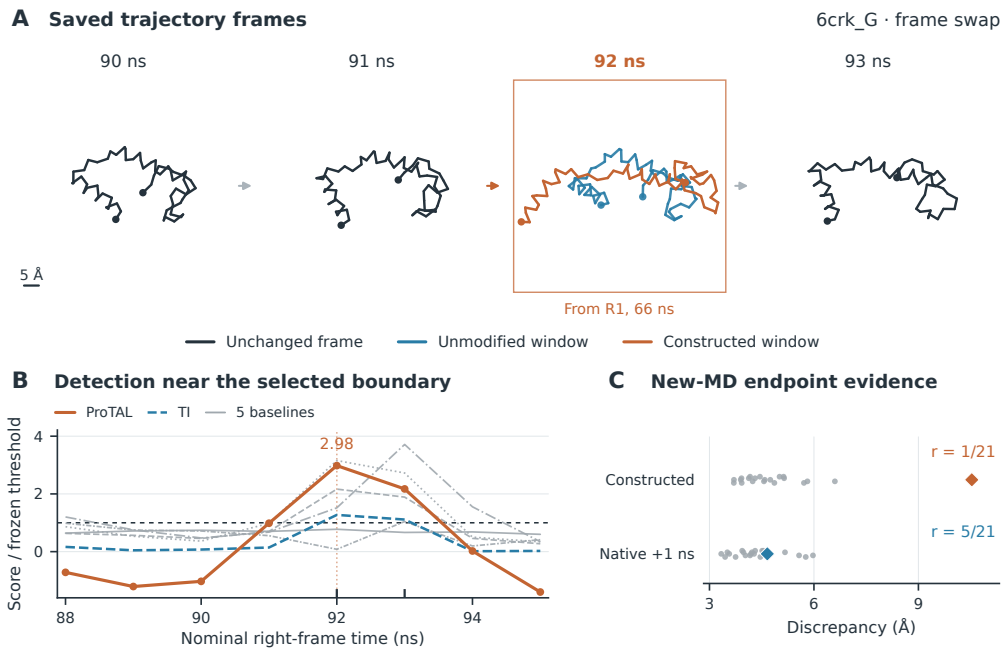


Figure 24: **S02: Shared W1 detection in `6crk_G` (frame swap). A:** aligned trajectory frames; **B:** normalized detection scores; **C:** conditional-MD endpoint discrepancies. The selected constructed transition joins R1 at 91 ns to R1 at 66 ns; the native control is R1 at 92 ns. The constructed and native targets have descriptive ranks $1/21$ and $5/21$, respectively. Panel definitions and the W1 criterion are given in Section C.16.

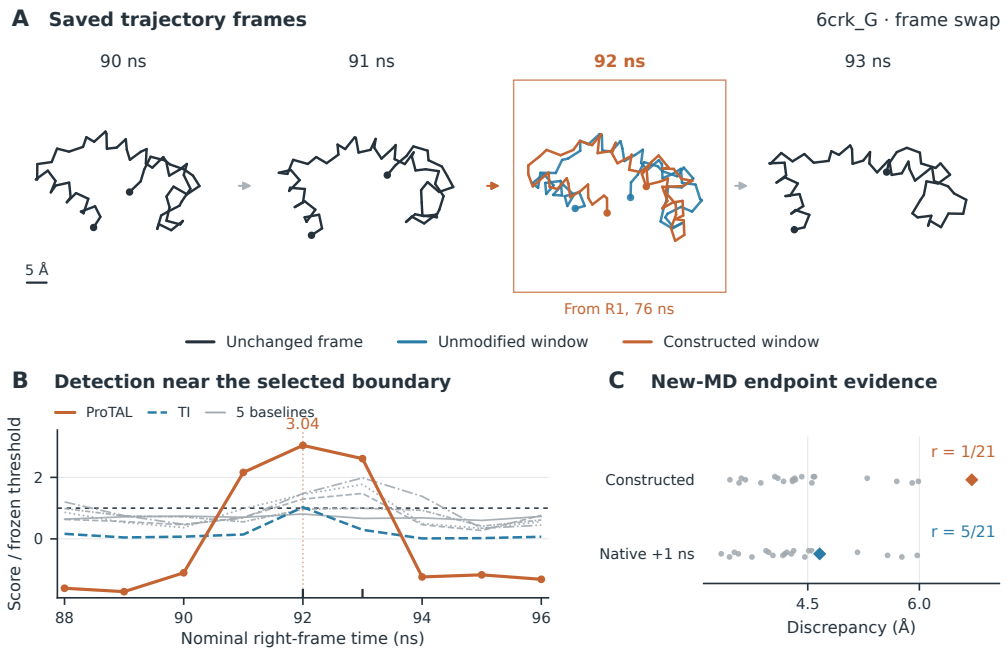


Figure 25: **S03: Shared W1 detection in `6crk_G` (frame swap). A:** aligned trajectory frames; **B:** normalized detection scores; **C:** conditional-MD endpoint discrepancies. The selected constructed transition joins R1 at 91 ns to R1 at 76 ns; the native control is R1 at 92 ns. The constructed and native targets have descriptive ranks $1/21$ and $5/21$, respectively. Panel definitions and the W1 criterion are given in Section C.16.

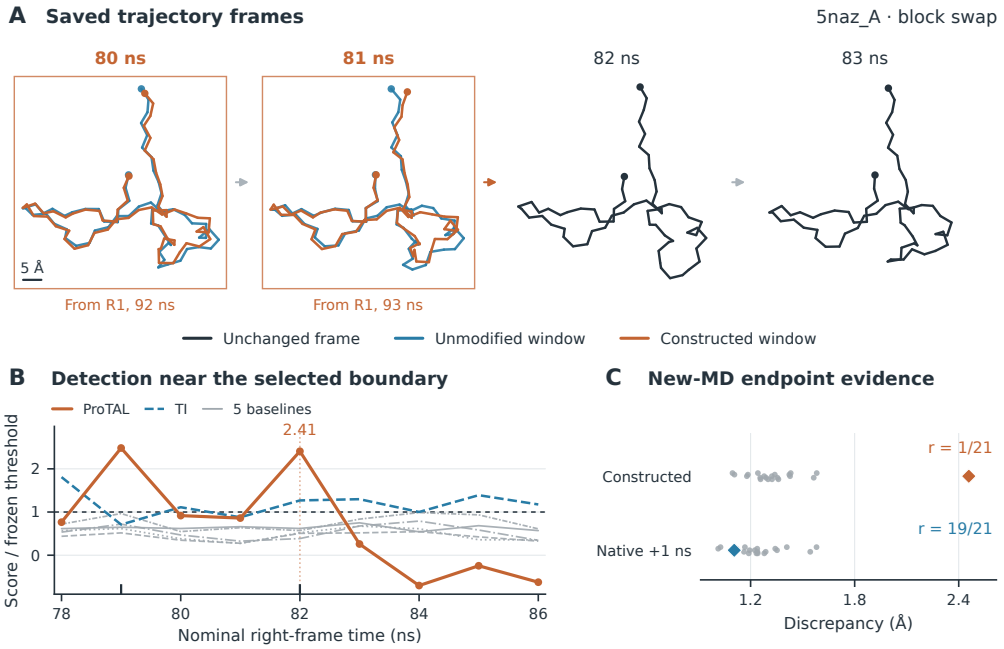


Figure 26: **S04: Shared W1 detection in `5naz_A` (block swap). A:** aligned trajectory frames; **B:** normalized detection scores; **C:** conditional-MD endpoint discrepancies. The selected constructed transition joins R1 at 93 ns to R1 at 82 ns; the native control is R1 at 94 ns. The constructed and native targets have descriptive ranks $1/21$ and $19/21$, respectively. The native control follows the actual replaced left frame and differs from the blue unmodified window in A. Panel definitions and the W1 criterion are given in Section C.16.

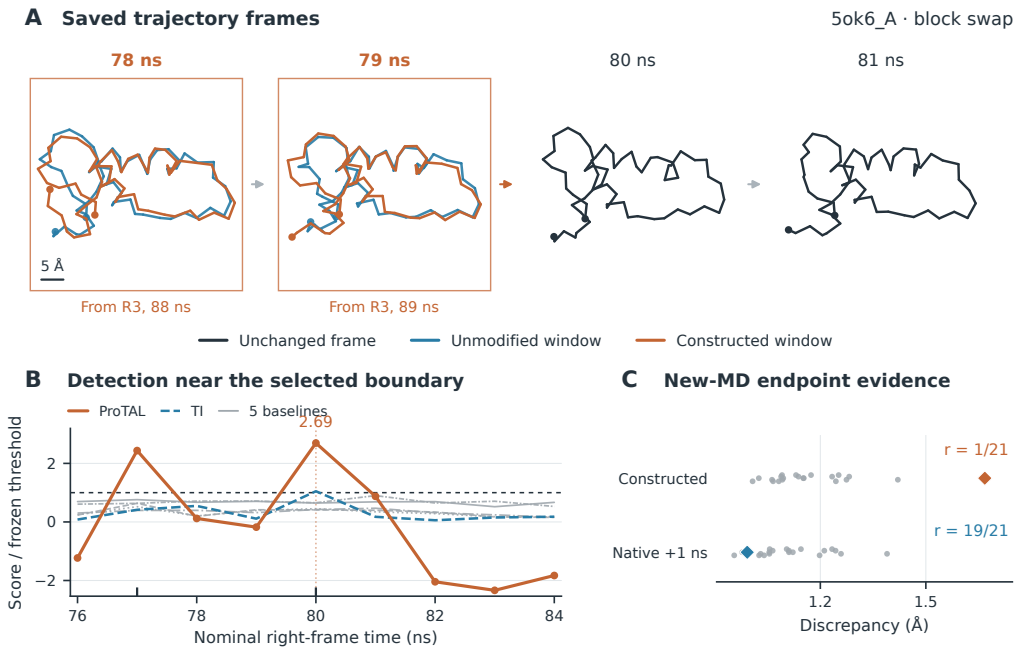


Figure 27: **S05: Shared W1 detection in `5ok6_A` (block swap). A:** aligned trajectory frames; **B:** normalized detection scores; **C:** conditional-MD endpoint discrepancies. The selected constructed transition joins R3 at 89 ns to R3 at 80 ns; the native control is R3 at 90 ns. The constructed and native targets have descriptive ranks $1/21$ and $19/21$, respectively. The native control follows the actual replaced left frame and differs from the blue unmodified window in A. Panel definitions and the W1 criterion are given in Section C.16.

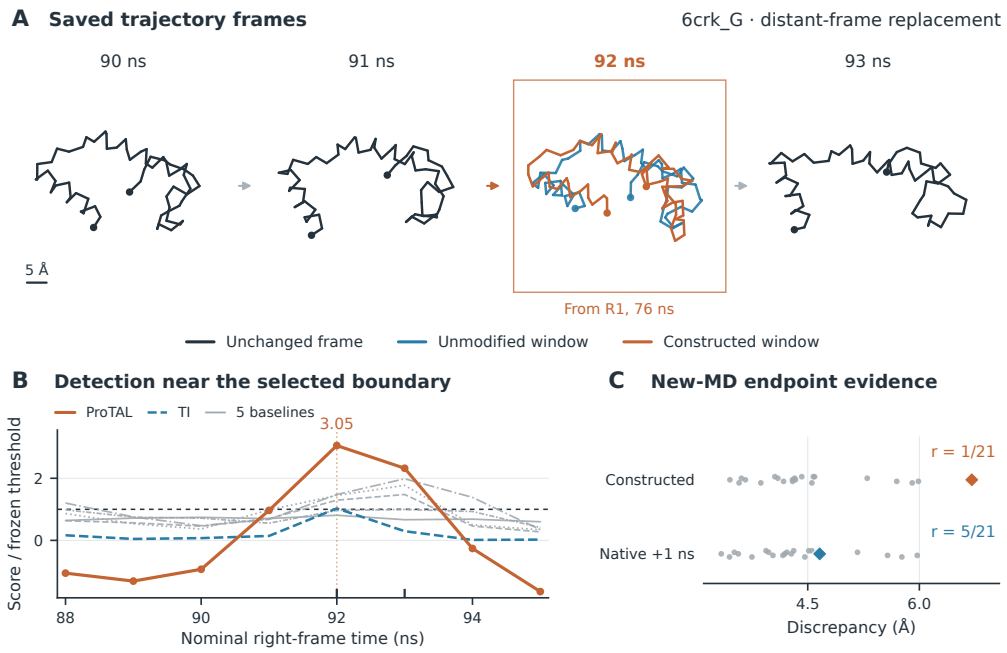


Figure 28: **S06: Shared W1 detection in `6crk_G` (distant-frame replacement). A:** aligned trajectory frames; **B:** normalized detection scores; **C:** conditional-MD endpoint discrepancies. The selected constructed transition joins R1 at 91 ns to R1 at 76 ns; the native control is R1 at 92 ns. The constructed and native targets have descriptive ranks $1/21$ and $5/21$, respectively. Panel definitions and the W1 criterion are given in Section C.16.

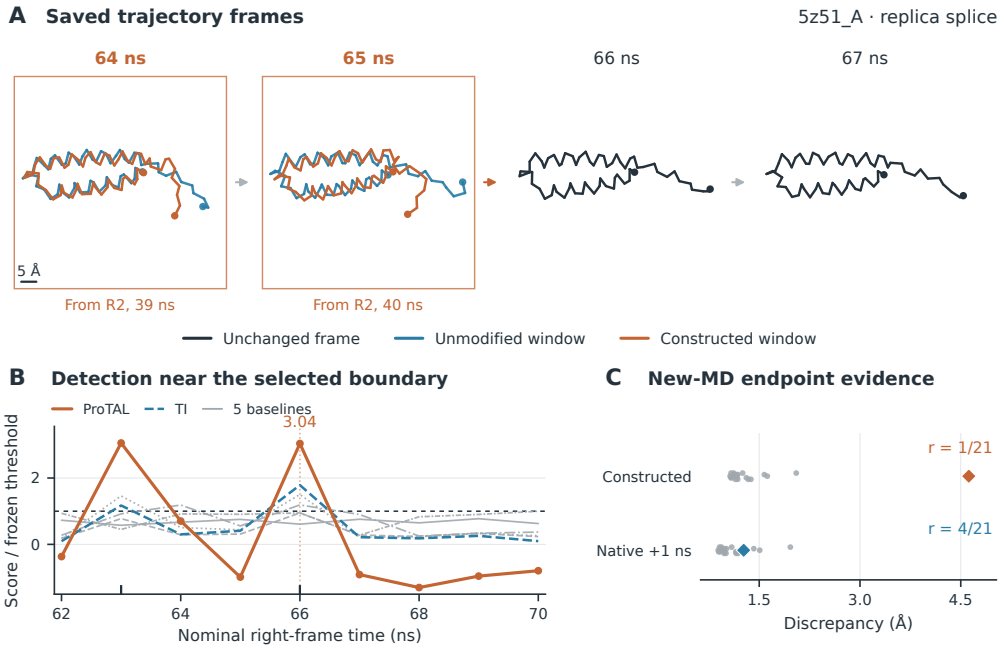


Figure 29: **S07: Shared W1 detection in `5z51_A` (replica splice). A:** aligned trajectory frames; **B:** normalized detection scores; **C:** conditional-MD endpoint discrepancies. The selected constructed transition joins R2 at 40 ns to R1 at 66 ns; the native control is R2 at 41 ns. The constructed and native targets have descriptive ranks $1/21$ and $4/21$, respectively. The native control follows the actual replaced left frame and differs from the blue unmodified window in A. Panel definitions and the W1 criterion are given in Section C.16.