WHERE THE RISK GOES: ERROR ALLOCATION IN LLM AGENTS UNDER CONFORMAL RISK CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

Conformal risk control can calibrate a commit-or-abstain rule so that an LLM agent's marginal probability of committing to a wrong answer stays below a chosen level. This guarantee does not control where those errors occur: the same overall risk can be distributed very differently across trajectories of different lengths. That distribution depends on the trajectory score the rule uses, while common summaries can obscure where uncertainty occurs or how uncertainty across steps combines. We introduce trajectory spatial contrast (TSC), a family of scores that treats the sequence of per-step uncertainty values as an ordered multivariate object. Each path is normalized to a fixed grid and scored by contrasting its spatial ranks against reference trajectories from correct and incorrect episodes. On synthetic examples designed to isolate trajectory structure, TSC separates failure patterns that no symmetric summary distinguishes and no additive score separates perfectly. Because every score is fitted on a separate reference split, conformal risk control gives TSC and competing scores the same marginal wrong-commit guarantee. Evaluating twelve scores across three benchmarks and three model families, we show that accumulating scores can concentrate almost all permitted wrong commits on the shortest trajectories, making the rule sensitive to shifts in the trajectory-length distribution. By contrast, the TSC family places errors most evenly across length groups in most benchmark–model pairs, at a modest cost in commit rate: it commits most in a minority of pairs and trails the best alternative by a few percentage points on average elsewhere.

1 INTRODUCTION

Language model-based agents solve tasks through sequences of reasoning steps, tool calls, and observations. These sequences, often referred to as trajectories or agent runs, provide information about how an agent arrives at its final answer. Uncertainty estimates collected throughout a trajectory, i.e. along steps, may therefore help assess the reliability of that answer beyond what can be inferred from the final step alone. This motivates using uncertainty estimates along a single trajectory to decide whether to accept the agent's final answer, namely, *commits*, or abstain.

A natural question is therefore how to accept as many answers as possible while controlling the probability of accepting an incorrect one. This involves two components: constructing a trajectory-level score from step-level uncertainty estimates, and calibrating a threshold below which answers are accepted. Given such a score, conformal risk control (CRC) calibrates the threshold to bound the overall probability of accepting an incorrect answer at a chosen level α (Angelopoulos et al., 2024; Abbasi-Yadkori et al., 2024). Under exchangeability, this guarantee holds in expectation over calibration and test draws. It concerns incorrect answers accepted per task (rather than the proportion of incorrect answers among those accepted). Importantly, the guarantee does not determine which answers are accepted, since that depends on the score. Hence, two scores calibrated to the same risk target may accept different numbers of answers and concentrate their accepted errors on different types of trajectories. Our focus is therefore on score design: how can we maintain a high acceptance rate while reducing the concentration of accepted errors?

Trajectory length introduces a particular challenge in constructing such a score. A long trajectory may reflect repeated failures, but it may also contain additional steps that help the agent reach a correct answer. Length can therefore be informative without being a reliable measure of uncertainty

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.

on its own. Yet, an aggregation rule can introduce a dependence on length independently of its relationship with correctness: summing nonnegative uncertainty contributions, for instance, penalizes additional steps even when they improve the final answer. Related length effects have been reported in uncertainty estimation for language models and LLM agents (Santilli et al., 2025; Matsnev, 2026). Hence, an acceptance rule based on such a score may consequently favor short trajectories, and its accepted errors can then concentrate among those trajectories, even while the overall risk remains controlled. The challenge is thus to distinguish useful information associated with trajectory length, from a mechanical penalty introduced by aggregation.

Simple summaries such as mean or maximum uncertainty avoid summing a penalty over every step, but discard the order in which uncertainty occurs. They assign the same score to sequences containing the same values, whether uncertainty occurs early or near the final answer. More broadly, existing work develops uncertainty signals for language models (Malinin & Gales, 2021; Duan et al., 2024; Vashurin et al., 2025) and methods for aggregating or propagating uncertainty across agent decisions (Zhao et al., 2025; Duan et al., 2025; Bouchard & Chauhan, 2026). Conformal abstention and ToolChain-CRC provide procedures for calibrating answer-acceptance or trajectory-acceptance rules (Abbasi-Yadkori et al., 2024; Opoku & Banahene, 2026). These methods provide uncertainty estimates and procedures for calibrating acceptance decisions, but a marginal risk guarantee alone does not determine how accepted errors are distributed across trajectory lengths. This motivates investigating scores that use the relative positions and joint configuration of uncertainty estimates, beyond simple summaries of their magnitude.

We introduce **trajectory spatial contrast (TSC)**, a trajectory-level uncertainty score that compares the ordered sequence of uncertainty estimates with labeled reference trajectories yielding correct and incorrect final answers. We first interpolate each sequence at fixed relative positions along the trajectory, obtaining vectors of a common dimension that retain coarse information about where uncertainty occurs. We then compute a spatial rank with respect to each reference group, using its norm as a measure of how central the query is within that group. The difference between the two norms defines a score in $[-1, 1]$. Lower scores favor trajectories that are more central among correct references than among incorrect ones. This construction compares the coordinates jointly, allowing the score to depend on their arrangement rather than only on separate step contributions. It has an explicit formula, requires no fitted density or reliability model, and has linear computational cost in the number of reference trajectories for a fixed representation dimension. We also consider a length-local variant, L-TSC, that compares a trajectory primarily with reference trajectories of similar length, by giving these references greater weight when computing each spatial rank. Both variants use labeled reference data separate from the calibration data, allowing their thresholds to be calibrated with the standard CRC marginal guarantee.

Our contributions are threefold. **(C1) A diagnosis of length-dependent error concentration.** We examine how accepted errors are distributed across short, medium, and long trajectories, measuring both each group's share of accepted errors and its probability of an accepted incorrect answer. We connect these patterns to accumulating scores and a length-only control, and quantify sensitivity to changes in the distribution of trajectory lengths. At $\alpha = 0.10$ with the primary uncertainty signal, a length-only score achieves 75–88% of the acceptance rate of pseudo-Noisy-OR on six of nine benchmark–model populations, while pseudo-Noisy-OR concentrates up to 100% of its observed accepted errors in the shortest group. **(C2) A score for ordered uncertainty sequences.** We develop spatial-rank contrasts with and without length weighting. On an explicit six-point construction, the contrast perfectly separates trajectories yielding correct and incorrect answers, whereas no permutation-invariant or thresholded coordinate-additive score does (Proposition A1). **(C3) An empirical comparison of acceptance rate and error concentration.** We compare twelve scores across three benchmarks and three model families using shared data splits, uncertainty signals, and a common CRC risk target. At $\alpha = 0.10$ with the primary signal, no single length group accounts for more than 55% of the accepted errors observed for L-TSC in any of the nine populations. Its acceptance rate is on average 3.1 percentage points below the best score for each population (Section 4). These findings show that score design can reduce error concentration while maintaining competitive acceptance rates.

2 TRAJECTORY SCORING AND CONFORMAL CALIBRATION

We first define the acceptance decision and the data used to construct and calibrate its score. We then introduce TSC and its length-weighted variant L-TSC, before describing the CRC procedure used to calibrate the acceptance threshold.

2.1 PROBLEM SETUP AND NOTATION

A task instance x with reference solution $y^\star$ is drawn from an unknown distribution. A frozen agent runs a ReAct loop (Yao et al., 2023): at step t it generates a free-text thought, possibly empty, and an action (a tool call or the final answer), and the environment returns an observation appended to its context, so that $s_t = (\text{thought}_t, \text{action}_t, \text{observation}_t)$. The agent need not be a reasoning model. Under native function calling (Patil et al., 2025), a step is one assistant response with the results of its calls (Appendix D). One episode or agent run is a trajectory $\tau = (s_1, \dots, s_T)$ of random length T, ending in a terminal candidate response $\hat{y}$. The task correctness is encoded as a binary label $E \in \{0, 1\}$, with $E = 1$ when the candidate is incorrect.

The framework considered in this paper is as follows: first, the agent's generated text and token log-probabilities are observable, but the agent cannot be re-prompted, resampled, or altered. Then, a *release gate* decides whether to release the final answer $\hat{y}$ or abstain. Hence, abstention does not undo previously executed tool calls (Liu et al., 2026); gating actions before execution (Ren et al., 2023; Feng et al., 2026b) is outside the scope of this paper. The tokens the agent generates at step t yield a scalar *step-level signal*

$$\phi_t \in [0, 1], \qquad t = 1, \dots, T, \tag{1}$$

oriented so that higher values indicate a more uncertain step. We leave its specific definition open here and describe the choices used in our experiments in Section 3. The sequence of step-level signals, $\Phi(\tau) = (\phi_1, \dots, \phi_T) \in [0, 1]^T$, is called the *signal path*; its dimension is the trajectory length T.

For a user-chosen risk level $\alpha \in (0, 1)$ we seek a rule $\mathsf{commit}(\tau) \in \{0, 1\}$ for any episode τ, with

$$\Pr\big(\mathsf{commit}(\tau) = 1,\ E = 1\big) \ \le\ \alpha. \tag{2}$$

We call the left-hand side the marginal wrong-commit rate (or marginal risk). The error rate among released outputs is this rate divided by the acceptance (or commit) rate $\Pr(\mathsf{commit} = 1)$ when the commit rate is positive; it is not controlled at level α. This risk counts accepted incorrect answers over all tasks; it differs from the error rate conditional on acceptance. Selective-risk control instead targets the error rate among released outputs, with high probability over the calibration draw (Geifman & El-Yaniv, 2017), and requires a different calibration objective.

The data are N tuples $(x_i, \tau_i, \hat{y}_i, y_i^\star)$ from independent task draws under the same frozen agent, partitioned independently of the data into a *reference* partition $\mathcal{I}_{\text{ref}}$ (size n_{ref}), a *calibration* partition $\mathcal{I}_{\text{cal}}$ (size n) and a *test* partition $\mathcal{I}_{\text{test}}$ (size n_{test}). The reference set is used to construct the score and choose any tuning parameters. Once constructed, the score is held fixed: the calibration set determines its acceptance threshold, and the test set is used to evaluate the resulting rule.

2.2 SPATIAL-RANK TRAJECTORY SCORES

Representing trajectories of different lengths. The step-level signals defined in Equation (1) form a sequence $\Phi(\tau) = (\phi_1, \dots, \phi_T)$ whose length T varies across trajectories. To compare these sequences, we first represent each of them by a vector with the same number of coordinates, which we call the *trajectory representation*. We fix $D \ge 2$ in advance, with $D = 4$ in our primary experiments.

For a trajectory with $T > 1$, we assign step t the relative position $(t-1)/(T-1)$, so that the first and last steps lie at positions 0 and 1. We linearly interpolate the signal values between consecutive steps and evaluate the resulting curve at D equally spaced positions:

$$\Phi^\star(\tau) = \text{Interp}_D\big(\Phi(\tau)\big) = \Big(\tilde{\varphi}_\tau\big(\tfrac{d-1}{D-1}\big)\Big)_{d=1}^{D} \in [0, 1]^D, \tag{3}$$

where $\tilde{\varphi}_\tau$ is the piecewise-linear interpolant through the points $\big((t-1)/(T-1), \phi_t\big)_{t=1}^{T}$. For $D = 4$, the coordinates give interpolated signal values at the beginning, one-third, two-thirds, and end of the

trajectory. When $T = 1$, we repeat the single signal value in all coordinates. This representation aligns trajectories by relative step position rather than absolute step number. It retains coarse information about the progression of the signal, but does not preserve every step or remove all information about length. Below, we will denote this interpolated trajectory representation by $U = \Phi^\star(\tau)$.

Comparing with correct and incorrect references. We construct a score using reference trajectories whose final-answer correctness is known. Their representations are divided into two groups:

$$C^+ = (U_j)_{j \in J^+}, \qquad C^- = (U_j)_{j \in J^-},$$

where $U_j = \Phi^\star(\tau_j)$, $J^+ = \{j \in \mathcal{I}_{\text{ref}} : E_j = 0\}$ contains the correct references, and $J^- = \{j \in \mathcal{I}_{\text{ref}} : E_j = 1\}$ contains the incorrect references. Repeated reference vectors are retained. We denote the corresponding class-conditional laws of U by P^+ and P^-.

To score a new trajectory τ, we compare its representation $U = \Phi^\star(\tau)$ with each reference group using spatial ranks (Möttönen & Oja, 1995; Chaudhuri, 1996; Conrad et al., 2026). For a nonempty group $C = (U_j)_{j \in J}$, the spatial rank is the average of the unit vectors pointing from each reference vector toward U. We use the norm of this average:

$$r(U; C) = \left\| \frac{1}{|J|} \sum_{j \in J} \frac{U - U_j}{\|U - U_j\|} \right\| \in [0, 1], \tag{4}$$

When $U = U_j$, the corresponding summand is defined as zero. Each nonzero summand has unit length, so the average reflects the directions of the reference vectors relative to U, rather than their distances from it. If the references surround U, these directions tend to cancel and $r(U; C)$ is small. If they lie predominantly on one side of U, the directions reinforce one another and $r(U; C)$ is larger. Thus, the rank norm describes how centrally the trajectory's representation lies within a reference group; it is not a probability of correctness.

The **trajectory spatial contrast (TSC)** score compares these two measurements:

$$s(\tau) = r(U; C^+) - r(U; C^-) \in [-1, 1], \qquad U = \Phi^\star(\tau). \tag{5}$$

A low score indicates that the trajectory's representation is more central among correct references than among incorrect ones. We accept the final answer when this score falls below a threshold, calibrated using the procedure in Section 2.3. The score is not a probability of correctness. Computing it requires comparing the representation with each reference vector, at a cost of $O(n_{\text{ref}} D)$ per trajectory.[1]

Length-weighted TSC. The score above compares a trajectory with all reference trajectories using equal weights within each group. We also consider a length-kernel variant, **local trajectory spatial contrast (L-TSC)**, that gives greater weight to references of similar length. This allows the comparison to account for the possibility that the same signal pattern has different meanings in short and long trajectories.

For a trajectory τ of length T, we assign reference trajectory j, of length T_j, the weight

$$\kappa_j(T) = \exp\left(-\frac{(\log T - \log T_j)^2}{2h^2} \right), \qquad w_j(T) = \frac{\kappa_j(T)}{\sum_{k \in J} \kappa_k(T)}.$$

These normalized Gaussian weights follow the Nadaraya–Watson construction (Nadaraya, 1964; Watson, 1964). Weights are normalized separately within each reference group $C = (U_j)_{j \in J}$. The parameter $h > 0$ determines how strongly length influences the comparison: smaller values concentrate weight on references closer in log length, while larger values yield more uniform weights.

We replace the equal weights in Equation (4) with these weights and compute the same difference between the two rank norms:

$$r_h(U, T; C) = \left\| \sum_{j \in J} w_j(T) \frac{U - U_j}{\|U - U_j\|} \right\|, \qquad s_h(\tau) = r_h(U, T; C^+) - r_h(U, T; C^-). \tag{6}$$

As in Equation (4), the summand is zero when $U = U_j$. We fix $h = 0.15$ in advance and examine other values in Figure 6. The resulting score is calibrated using one global acceptance threshold, rather than a separate threshold for each length. Its effect on error concentration is evaluated empirically (Section 4); the variant does not provide a risk guarantee conditional on trajectory length.

[1] If either reference group is empty, the rule abstains on all tasks.

2.3 CALIBRATING THE ACCEPTANCE THRESHOLD

Once a trajectory score s has been constructed using the reference data, we keep it fixed and use the calibration set to choose an acceptance threshold. For a threshold λ, the rule accepts the final answer when $s(\tau) < \lambda$. Increasing λ accepts more answers, but may also accept more incorrect ones. We use conformal risk control (CRC) (Angelopoulos et al., 2024) to select the threshold under a prescribed risk target α. The resulting rule is $\mathsf{commit}(\tau) = \mathbb{1}\{s(\tau) < \hat{\lambda}\}$.

For calibration trajectory $i \in \mathcal{I}_{\text{cal}}$, let $s_i = s(\tau_i)$. Since $E_i = 1$ denotes an incorrect answer, the loss

$$L_i(\lambda) \;=\; E_i \cdot \mathbb{1}\{s_i < \lambda\}, \qquad \lambda \in \overline{\mathbb{R}}, \tag{7}$$

equals one exactly when the rule accepts an incorrect answer, and zero otherwise. This is the abstention loss used by Abbasi-Yadkori et al. (2024); the accepted-trajectory loss of Opoku & Banahene (2026) has the same form, with an audited trajectory-failure label in place of E_i. The loss is non-decreasing in λ, left-continuous, bounded by one, and zero at $\lambda = -\infty$, as required by CRC.

Averaging over all n calibration trajectories gives the empirical risk $\widehat{R}_n(\lambda) = n^{-1} \sum_{i \in \mathcal{I}_{\text{cal}}} L_i(\lambda)$. CRC uses the finite-sample correction $(n\widehat{R}_n(\lambda) + 1)/(n + 1)$ and sets

$$\hat{\lambda} \;=\; \sup\Big\{\lambda \in \overline{\mathbb{R}} : \tfrac{n\widehat{R}_n(\lambda)+1}{n+1} \le \alpha\Big\}, \tag{8}$$

the $(q{+}1)$-th smallest score among the incorrect calibration episodes, $q = \lfloor \alpha(n{+}1) \rfloor - 1$ (infinite thresholds at the boundaries: Theorem A2, Appendix B).

Assumption 1. *Conditional on the reference data $\{(\tau_j, \hat{y}_j, y_j^\star) : j \in \mathcal{I}_{\text{ref}}\}$, the tuples $(x_i, \tau_i, \hat{y}_i, y_i^\star)$, $i \in \{1, \ldots, n, n{+}1\}$, are exchangeable; indices $1, \ldots, n$ denote calibration data and $n{+}1$ denotes the test example.*

This assumption concerns entire trajectories, rather than individual steps. Steps within a trajectory may therefore be dependent; exchangeability is required across the calibration and test trajectories, conditional on the reference data.

Under Assumption 1, for any such score and any $\alpha \in (0, 1)$, the threshold in Equation (8) ensures

$$\Pr\big(\mathsf{commit}(\tau_{n+1}) = 1,\; E_{n+1} = 1\big) \;\le\; \alpha, \tag{9}$$

where the probability is taken jointly over calibration and test draws, conditional on the reference data (Angelopoulos et al., 2024). The guarantee holds separately for each score constructed from the reference data; it does not bound the error rate conditional on acceptance.

3 EXPERIMENTAL SETUP

The goal of our experiments is to assess how trajectory scoring affects both the proportion of answers accepted and the concentration of accepted errors across trajectory lengths. This section describes the benchmarks and models, discusses the choice of step-level signals in the framework of Section 2.1, and introduces the alternative trajectory scores evaluated alongside TSC and L-TSC (Section 2.2). Acceptance thresholds are calibrated using the procedure in Section 2.3. We then define the metrics used to evaluate acceptance and error concentration. Further details on the datasets, step-level signals, and evaluation protocols are provided in Appendix D.

Table 1: **Benchmarks, correctness criteria, and data splits.** Episode counts are given per model; split sizes are listed in reference/calibration/test order. BFCL's four multi-turn categories are pooled.

Benchmark	Number of Episodes	Correctness	Protocol	ref/cal/test
MuSiQue (Trivedi et al., 2022)	2,417	F1 ≥ 0.5 against gold	rotation	725/846/846
BFCL v3 multi-turn (Patil et al., 2025)	800	state-and-response checker	rotation	240/280/280
WebShop (Yao et al., 2022)	1,000 dev. + 500 test	reward $= 1$ (strict success)	held-out	300/700/500

Datasets, models and correctness metrics. We consider three models: Gemma 3 12B, Mistral Small 4, and Muse Glimmer 30B (Gemma Team, 2025; Mistral AI, 2026; Meta Superintelligence Lab, 2026). Each model is evaluated on three benchmarks, yielding nine benchmark–model populations. Within each population, all trajectory scoring methods use the same generated trajectories and step-level signals. Table 1 summarizes the benchmarks, the criteria used to determine whether a response is correct, and the sizes of the reference, calibration, and test sets.

The Protocol column in Table 1 specifies how the data splits are formed across 50 repetitions. For MuSiQue and BFCL, *rotation* keeps the reference set fixed and randomly divides the remaining episodes into calibration and test halves at each repetition. For WebShop, *held-out* keeps the test set fixed and randomly divides the development episodes into reference and calibration sets at each repetition. We assume that trajectories within each benchmark–model population satisfy the exchangeability condition in Assumption 1; for WebShop, this assumption also concerns the development and held-out test data and is not ensured by random splitting alone.

Step-level uncertainty signals. Our main step-level signal, denoted E1, is one minus the geometric mean of the probabilities assigned by the model to the tokens generated at that step. Higher values therefore indicate lower confidence in the generated text. To assess sensitivity to this choice, we repeat the full grid under mean token entropy (E3) (Malinin & Gales, 2021). All signals follow the convention in Equation (1): values lie in $[0, 1]$ (using increasing transformations where needed), with larger values indicating greater uncertainty. Signals include the terminal step.

Trajectory scoring methods. Given a sequence of step-level signals, a trajectory scoring method computes a scalar score that will be used to decide whether to accept the final answer or abstain. Alongside TSC and L-TSC (Section 2.2), we consider the following alternatives. The simplest methods use the mean or maximum of the step-level signals. Another uses only the final-step signal, inspired by UALA (Han et al., 2024). Product-based scores include pseudo-Noisy-OR (Opoku & Banahene, 2026), which computes one minus the product of the complements of step uncertainties, and its length-normalized variant (product aggregation of verbalized step confidences; Zhang et al., 2026b; Matsnev, 2026). RMS (Zhao et al., 2025) computes the root mean square of step uncertainties, while our positional SAUP variant (Zhao et al., 2025) uses a weighted RMS that weights later steps more heavily. The aggregation adapted from TRACER (Tayebati et al., 2026) combines the average of the most uncertain steps with the maximum step uncertainty. Worst-step Bonferroni compares the most uncertain step with uncertainty values from correct reference trajectories. As an additional acceptance rule, coordinatewise Bonferroni calibrates each of D interpolated coordinates at level α/D and accepts if any coordinate falls below its threshold. Finally, a *length-only gate* serves as a control, using $\log T$ as its score without reading the step-level signals. Detailed definitions and adaptations are provided in Appendix C.

Comparison on a common trajectory representation. The scoring methods above differ both in how they represent the sequence of step-level signals and in how they compute a score. To compare how scoring functions use the same representation of each test trajectory, we conduct an additional study using the interpolated vector $U = \Phi^\star(\tau) \in [0, 1]^4$ defined in Section 2.2, with E1 as the step-level signal. We apply six scoring functions to this vector: mean, max, RMS, pseudo-Noisy-OR, the final-coordinate score, and TSC. No scoring method receives trajectory length explicitly. Each score has its own threshold, calibrated at the same target risk level using the same procedure.

Metrics – Acceptance and wrong-commit rates. We evaluate how often each method accepts a final answer when calibrated at the same nominal risk level α, using matched data splits. We report the acceptance (or commit) rate CR and the empirical wrong-commit rate CER, defined as the proportion of test episodes whose final answer is both accepted and incorrect. The latter estimates the marginal wrong-commit rate defined in Equation (2). Our main experiments use $\alpha = 0.10$. Although all methods share the same calibration target, their empirical wrong-commit rates can differ.

Metrics – Error allocation across trajectory lengths. To examine whether wrong commits concentrate on trajectories of particular lengths, we group test episodes by trajectory length and report two complementary metrics. Group boundaries are defined by length terciles within each benchmark–model population, using the full episode pool for MuSiQue and BFCL and the development set for WebShop. These groups need not contain equal numbers of test episodes.

Let $p_g = \Pr(G = g)$ be the proportion of episodes in group g and $r_g = \Pr(\mathsf{commit} = 1 \wedge \hat{y} \neq y^\star \mid G = g)$ its wrong-commit rate. The marginal wrong-commit rate is then $R = \sum_g p_g r_g$. We report

$$W = \max_g r_g, \qquad H = \max_g \frac{p_g r_g}{R} \quad (R > 0).$$

The *highest within-group wrong-commit rate* W measures how often wrong commits occur in the most affected length group. The *largest share of wrong commits* H measures the largest fraction of all wrong commits contributed by any one group. Note that neither metric is controlled by the marginal risk guarantee in Section 2.3, which applies to the overall wrong-commit rate R. We estimate both metrics from the test episodes over the 50 random data splits described earlier in this section.

Estimation and variability. We conclude this section by describing how scores are fitted and variability is assessed. All score fitting and parameter tuning use only the reference set defined in Section 2.1. Methods are evaluated on the same data splits and calibrated at the same target risk level α. We report 95% bootstrap confidence intervals to quantify variability across the 50 random data splits. These intervals describe variation across splits of the observed episodes, rather than newly generated trajectories. See Appendix D for details.

4 RESULTS

We first report empirical wrong-commit rates and acceptance rates, then examine how wrong commits are distributed across trajectory lengths. We next compare scoring functions on a common trajectory representation. Unless otherwise stated, results use E1 and $\alpha = 0.10$. Table 2 presents a subset of six trajectory scoring methods, including TSC and L-TSC. Results for all twelve methods are reported in Table 8 in Appendix E.

Empirical wrong-commit rates. Across the nine populations, the empirical wrong-commit rate CER of every non-degenerate score, averaged over the 50 data splits, is at most 10.5% at the target level $\alpha = 0.10$ (Figure 3 and Table 8). These observations are consistent with the marginal risk guarantee in Equation (9), which holds in expectation over calibration and test draws under exchangeability for each pre-specified score.

Acceptance rates. We first examine the acceptance (or commit) rates, denoted CR, reported in Table 2. No score achieves the highest acceptance rate on every population. For example, the final-step score leads on MuSiQue with Gemma 3, whereas L-TSC leads on WebShop with Gemma 3 and Mistral Small 4. Averaged over the nine populations, pseudo-Noisy-OR has the smallest acceptance-rate gap, defined as the difference between the highest acceptance rate in each population and that of the method: 0.9 percentage points. The corresponding gaps for TSC and L-TSC are 2.9–3.1 points (Table 11). Thus, TSC and L-TSC accept, on average, about three percentage points fewer answers than the best method in each population. We next examine the associated concentration of wrong commits across length groups.

The role of trajectory length. To examine the role of length, we use the results by length group in Table 10 (Appendix E), which also include the length-only gate. At constant step-level signal u, the pseudo-Noisy-OR score is $1 - (1 - u)^T$, which increases with trajectory length T. A low-score acceptance rule therefore favors shorter trajectories even when their step-level signals are identical. We then examine where the accepted errors occur: on seven of the nine populations, at least 91% of the wrong commits made by pseudo-Noisy-OR fall in the shortest length group.

To assess how much of the acceptance rate can be achieved using length alone, we consider the length-only gate, which scores trajectories by $\log T$ without using the step-level signals. On six of the nine populations, the ratio of the acceptance rate of the length-only gate to that of pseudo-Noisy-OR lies between 0.75 and 0.88. However, length alone is less effective on the remaining three populations. On MuSiQue with Gemma 3, the length-only gate accepts less than half as many answers as pseudo-Noisy-OR. On WebShop with Gemma 3 or Mistral Small 4, it accepts none. Thus, length alone can yield a substantial acceptance rate on several populations, but the step-level signals allow pseudo-Noisy-OR to accept additional answers under the same calibration target.

Wrong commits across trajectory lengths. We next examine whether TSC and L-TSC reduce the concentration of wrong commits across length groups. We use the full comparison in Appendix E

Table 2: **Commit rates and wrong-commit concentration at $\alpha = 0.10$ (E1).** Six scores of the full grid (Table 8) on the nine benchmark–model populations: commit rate CR, highest within-group wrong-commit rate W and largest single-group share of wrong commits H (%). CR: mean ± std over 50 splits; W, H: ± half-width of a 95% bootstrap interval. Bold and underline: best and second-best in the full grid, ties included. † our primary configuration (TSC without length weighting, $D = 4$).

MuSiQue	**Gemma 3 12B**			**Mistral S4**			**Muse Glimmer 30B**		
Score	**CR↑**	**W↓**	**H↓**	**CR↑**	**W↓**	**H↓**	**CR↑**	**W↓**	**H↓**
mean step score	24.8 ± 2.0	16.0±0.6	70±1	18.3 ± 2.0	24.0±0.8	80±1	14.4 ± 1.7	21.2±0.7	74±1
max step score	22.7 ± 2.2	19.6±0.7	85±1	26.8 ± 2.0	14.7±0.5	51±1	16.8 ± 1.6	11.8±0.5	39±1
pseudo-Noisy-OR	23.8 ± 1.9	21.5±0.8	94±1	**28.8 ± 2.1**	26.4±0.7	95±1	**20.6 ± 2.1**	28.5±0.9	95±1
final-step score (UALA-style)	**26.8 ± 2.2**	12.5±0.6	54±1	21.0 ± 2.6	19.5±0.7	64±1	13.3 ± 2.0	18.9±0.7	65±1
TSC (D=4)†	23.7 ± 2.6	17.7±0.6	77±1	23.3 ± 2.5	15.0±0.8	49±1	17.8 ± 1.9	**11.4±0.5**	**38±1**
L-TSC (h=0.15)	25.0 ± 2.2	**11.7±0.7**	**48±1**	22.8 ± 2.8	**10.2±0.6**	**34±1**	19.9 ± 2.4	12.1±0.5	42±1
BFCL pooled	**Mistral S4**			**Gemma 3 12B**			**Muse Glimmer 30B**		
Score	**CR↑**	**W↓**	**H↓**	**CR↑**	**W↓**	**H↓**	**CR↑**	**W↓**	**H↓**
mean step score	18.6 ± 3.8	12.0±1.0	44±2	12.1 ± 2.9	14.1±1.1	45±2	22.6 ± 4.2	19.9±1.4	58±2
max step score	19.0 ± 2.8	15.6±0.9	62±2	**13.0 ± 2.5**	12.4±1.0	48±2	18.5 ± 3.9	**11.2±1.2**	**34±1**
pseudo-Noisy-OR	18.0 ± 3.7	23.8±1.6	91±1	12.9 ± 2.7	19.5±1.3	75±2	**30.5 ± 4.3**	23.8±1.5	84±2
final-step score (UALA-style)	20.3 ± 3.9	**11.2±1.2**	39±2	10.4 ± 2.5	14.1±0.9	55±2	20.6 ± 3.0	22.6±1.5	64±2
TSC (D=4)†	**20.9 ± 3.2**	12.1±1.2	40±2	12.7 ± 3.3	11.3±1.0	**36±2**	21.7 ± 3.3	16.1±1.4	48±2
L-TSC (h=0.15)	20.1 ± 3.5	13.2±0.9	50±2	11.5 ± 2.9	12.4±1.1	48±2	19.3 ± 3.0	16.9±0.9	48±2
WebShop (held-out test)	**Gemma 3 12B**			**Mistral S4**			**Muse Glimmer 30B**		
Score	**CR↑**	**W↓**	**H↓**	**CR↑**	**W↓**	**H↓**	**CR↑**	**W↓**	**H↓**
mean step score	8.3 ± 0.4	25.6±0.1	89±1	6.0 ± 0.6	19.0±0.5	100±0	11.6 ± 0.4	18.1±0.1	74±0
max step score	13.5 ± 0.8	17.1±0.3	87±0	13.5 ± 0.5	20.0±0.2	80±0	19.3 ± 1.5	16.5±0.4	72±0
pseudo-Noisy-OR	13.0 ± 1.0	20.0±0.5	100±0	14.2 ± 1.1	30.4±0.8	98±0	**23.3 ± 1.7**	27.4±0.7	98±0
final-step score (UALA-style)	8.7 ± 0.3	31.6±0.3	100±0	6.2 ± 0.4	19.6±0.4	100±0	15.8 ± 0.9	19.8±0.2	66±0
TSC (D=4)†	12.4 ± 1.5	15.3±0.7	73±3	14.0 ± 1.5	**12.8±0.5**	**46±1**	20.1 ± 2.2	**10.6±0.7**	**40±2**
L-TSC (h=0.15)	**14.1 ± 1.7**	**10.9±1.0**	**55±5**	**14.8 ± 1.8**	17.1±1.4	55±4	17.6 ± 2.5	11.2±1.3	43±4

(Table 8), considering both the largest share of wrong commits in one group, H, and the highest within-group wrong-commit rate, W. Among all eleven non-degenerate scoring rule methods, at least one of TSC and L-TSC achieves the lowest H on seven of the nine populations and the lowest W on six. For example, on MuSiQue with Mistral Small 4, L-TSC has $H = 34\%$, compared with 51% for the best score outside the TSC family and 95% for pseudo-Noisy-OR. In this population, the wrong commits made by pseudo-Noisy-OR occur almost exclusively on short trajectories. With L-TSC, they are spread across short, medium, and long trajectories, with no length group accounting for most of them. This describes where the errors occur, rather than a reduction in their overall frequency. It also bounds what a change in the length mixture can cost: if the within-group laws are unchanged, the wrong-commit rate under any mixture within total-variation distance ρ of the source is at most $\alpha + \rho\,(\max_g r_g - \min_g r_g)$ (Corollary A1, Appendix B). Tables 11 and 13 evaluate this bound at $\rho = 0.2$, where TSC and L-TSC have the lowest worst case over the nine populations. Taken together, these results show that the TSC family offers a useful trade-off: TSC and L-TSC achieve acceptance rates within about three percentage points of the best method per population on average, while attaining the lowest error concentration on most populations.

Comparison on a common trajectory representation. We next compare scoring functions applied to the same four-coordinate representation, using the study described in Section 3. Figure 1 reports the differences between TSC and two aggregations of this vector: pseudo-Noisy-OR and the final-coordinate score. Positive differences in CR and negative differences in W, H favor TSC.

Against either comparator, TSC achieves a higher commit rate on seven of the nine populations. On six of these, it also achieves lower W and H: it accepts more answers while reducing both the highest within-group wrong-commit rate and the concentration of wrong commits in a single group. The improvement is not universal: on MuSiQue with Gemma 3, the final-coordinate score

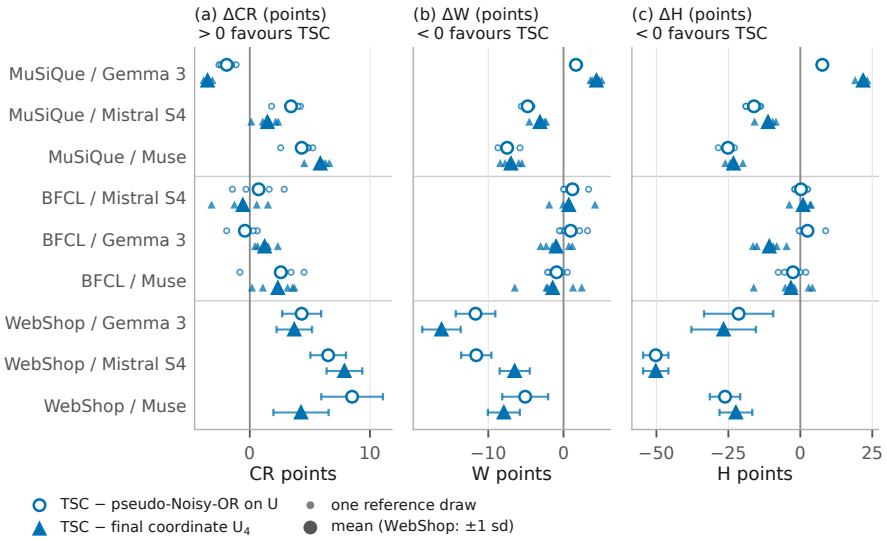

Figure 1: **Trajectory scores compared on a common four-coordinate representation (E1, $\alpha = 0.10$).** Differences in points between TSC and pseudo-Noisy-OR (open circles) or the final-coordinate score (filled triangles); positive $\Delta\mathrm{CR}$ and negative ΔW, ΔH favor TSC. Small symbols: five reference sets; large: their mean (WebShop: mean ± 1 std over 50 development splits, test set fixed). Details in Appendix D.

performs better on all three metrics. The final-coordinate score is identical to the final-step score in Table 2, since $U_4 = \phi_T$. For pseudo-Noisy-OR, however, the comparison changes: it now combines four interpolated values, regardless of trajectory length, rather than all T step-level signals as in Table 2. The sensitivity of TSC and L-TSC to the two parameters introduced in Section 2.2, the number of coordinates D and the bandwidth h, and further experiments with alternative trajectory representations are reported in Appendix E.

5 Conclusion

Under a marginal error budget, where the accepted errors land depends on how uncertainty is aggregated across steps. TSC compares the ordered uncertainty sequence with labeled reference clouds, a spatial-rank contrast that fits no parametric density or reliability model. On an exact construction it separates what no symmetric summary distinguishes and no additive score separates perfectly. Under one calibration and cap, TSC is ahead of the aggregations of the same coordinates on most populations; TSC or L-TSC places the permitted errors more evenly than every other score of the grid on seven of the nine populations, about three points of commit rate behind the per-population leader on average, and ahead on both axes on one population. Both variants also have the lowest worst case over the nine populations under a bounded shift of the length mixture (Corollary A1). In deployment, an unattended agent over long tool-call chains that meets its cap while concentrating its wrong commits in one length group fails silently there.

Limitations and future work. The evidence spans three benchmarks, three model families, one generation per task, and token-distribution signals; the comparison set is the grid of scores in use, and the WebShop intervals are pointwise and conditional on the realized development draws. The certificate is marginal, in expectation over calibration and test draws, per pre-specified score: one global calibration, a bound on wrong commits per episode and not on the error among released outputs, and no per-group control. Future work should examine the black-box setting, where the token distribution is unavailable and the score must rest on verbalized or API-level signals, and the calibration layer, held fixed here at one global CRC threshold and a problem in its own right.

AI USE STATEMENT

We used a large-language-model coding assistant, under author direction, for manuscript drafting and editing, for proofs, for analysis and pipeline code, for feedback on experiments and the interpretation of results, and for literature-search support. We have not used generative AI tools for research ideation or the generation of any experimental data or result. All AI-assisted text and code were reviewed by the authors; every number in the paper is produced by scripts that read the archived experiment outputs, and AI-suggested citations were verified against the cited papers. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Assumptions and complete proofs of all stated results are given in Appendix B. Appendix C states the implemented formula of every baseline and discloses each deviation from its published form. Appendix D specifies the trajectory populations, uncertainty signals, calibration/test splits and the representation-study protocol. The supplementary material provides the source code, the configurations, the trajectory and signal caches with their provenance manifest, and the scripts that regenerate every results table and figure from them, together with the generation scripts. Six of the nine populations were generated on a single desktop-class node (one NVIDIA DGX Spark), and calibration and evaluation need no GPU: the whole certificate of one split, reference fit included, takes at most 0.13 s on one CPU thread (Appendix D).

ETHICS STATEMENT

This work builds machinery for *withholding* agent outputs whose error rate cannot be certified. The guarantee is marginal and in expectation; it supplies no per-group or per-instance assurance (e.g. fairness across user populations), as Section 4 illustrates for length groups. The rule gates the terminal output after the episode: it does not prevent side effects of tool calls the agent has already executed (Section 2.1). No human subjects or private data are involved; all benchmarks are public.

REFERENCES

Yasin Abbasi-Yadkori, Ilja Kuzborskij, David Stutz, András György, Adam Fisch, Arnaud Doucet, Iuliya Beloshapka, Wei-Hung Weng, Yao-Yuan Yang, Csaba Szepesvári, Ali Taylan Cemgil, and Nenad Tomasev. Mitigating LLM Hallucinations via Conformal Abstention. arXiv preprint arXiv:2405.01563, 2024. URL https://arxiv.org/abs/2405.01563.

Lukas Aichberger, Kajetan Schweighofer, and Sepp Hochreiter. Rethinking Uncertainty Estimation in LLMs: A Principled Single-Sequence Measure. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=wdhruVcRx1.

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *International Conference on Learning Representations (ICLR)*, 2024.

Tian Bai and Ying Jin. Conformal Selective Prediction with General Risk Control. arXiv preprint arXiv:2603.24704, 2026. URL https://arxiv.org/abs/2603.24704.

Konstantina Bairaktari, Jiayun Wu, and Steven Wu. Kandinsky Conformal Prediction: Beyond Class- and Covariate-Conditional Coverage. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 2581–2602. PMLR, 2025. URL https://proceedings.mlr.press/v267/bairaktari25a.html.

Stephen Bates, Anastasios N. Angelopoulos, Lihua Lei, Jitendra Malik, and Michael I. Jordan. Distribution-free, risk-controlling prediction sets. *Journal of the ACM*, 68(6):1–34, 2021.

Vincent Blot, Anastasios N. Angelopoulos, Michael I. Jordan, and Nicolas J.-B. Brunel. Automatically adaptive conformal risk control. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 258 of *Proceedings of Machine Learning Research*, pp. 19–27. PMLR, 2025.

Dylan Bouchard and Mohit Singh Chauhan. Beyond Single-Turn Confidence: Trajectory-Adapted Uncertainty Quantification for LLM Agents. arXiv preprint arXiv:2608.11552, 2026. URL https://arxiv.org/abs/2608.11552.

Kwan Ho Ryan Chan, Yuyan Ge, Edgar Dobriban, Hamed Hassani, and René Vidal. Conformal Information Pursuit for Interactively Guiding Large Language Models. In *Advances in Neural Information Processing Systems*, volume 38, pp. 100312–100361, 2025. doi: 10.52202/085713-3021. URL https://papers.nips.cc/paper_files/paper/2025/hash/823a3d2cf462fb815978314023e48f65-Abstract-Conference.html.

Probal Chaudhuri. On a geometric notion of quantiles for multivariate data. *Journal of the American Statistical Association*, 91(434):862–872, 1996.

Catherine Chen, Jingyan Shen, Zhun Deng, and Lihua Lei. Conformal Tail Risk Control for Large Language Model Alignment. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 8955–8978. PMLR, 2025. URL https://proceedings.mlr.press/v267/chen25bd.html.

Tiejin Chen, Huaiyuan Yao, Jia Chen, Evangelos E. Papalexakis, and Hua Wei. Every Response Counts: Quantifying Uncertainty of LLM-based Multi-Agent Systems through Tensor Decomposition. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16204–16218. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.737. URL https://aclanthology.org/2026.acl-long.737/.

John J. Cherian, Isaac Gibbs, and Emmanuel J. Candès. Large language model validity via enhanced conformal prediction methods. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/d02ff1aeaa5c268dc34790dd1ad21526-Abstract-Conference.html.

Victor Chernozhukov, Alfred Galichon, Marc Hallin, and Marc Henry. Monge–Kantorovich depth, quantiles, ranks and signs. *The Annals of Statistics*, 45(1):223–256, 2017.

Anton Conrad, Eric Moulines, and Julien Perez. Geometric conformal prediction with spatial ranks and multivariate quantiles. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=C9SwK10bTT.

Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. Shifting Attention to Relevance: Towards the Predictive Uncertainty Quantification of Free-Form Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5050–5063, 2024. doi: 10.18653/v1/2024.acl-long.276. URL https://aclanthology.org/2024.acl-long.276/.

Jinhao Duan, James Diffenderfer, Sandeep Madireddy, Tianlong Chen, Bhavya Kailkhura, and Kaidi Xu. UProp: Investigating the Uncertainty Propagation of LLMs in Multi-Step Agentic Decision-Making. arXiv preprint arXiv:2506.17419, 2025. URL https://arxiv.org/abs/2506.17419.

Subhajit Dutta, Soham Sarkar, and Anil K. Ghosh. Multi-scale Classification using Localized Spatial Depth. *Journal of Machine Learning Research*, 17(217):1–30, 2016. URL https://jmlr.org/papers/v17/16-113.html.

Ekaterina Fadeeva, Roman Vashurin, Akim Tsvigun, Artem Vazhentsev, Sergey Petrakov, Kirill Fedyanin, Daniil Vasilev, Elizaveta Goncharova, Alexander Panchenko, Maxim Panov, Timothy Baldwin, and Artem Shelmanov. LM-Polygraph: Uncertainty Estimation for Language Models. In Yansong Feng and Els Lefever (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 446–461, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-demo.41. URL https://aclanthology.org/2023.emnlp-demo.41/.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630:625–630, 2024. doi: 10.1038/s41586-024-07421-0. URL https://www.nature.com/articles/s41586-024-07421-0.

Naihe Feng, Yi Sui, Shiyi Hou, Ga Wu, and Jesse C. Cresswell. Conformal Agent Error Attribution. arXiv preprint arXiv:2605.06788, 2026a. URL https://arxiv.org/abs/2605.06788.

Yushi Feng, Junye Du, Qifan Wang, Zizhan Ma, Qian Niu, Yutaka Matsuo, Long Feng, and Lequan Yu. CORA: Conformal Risk-Controlled Agents for Safeguarded Mobile GUI Automation. arXiv preprint arXiv:2604.09155, 2026b. URL https://arxiv.org/abs/2604.09155.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Gemma Team. Gemma 3 Technical Report. arXiv preprint arXiv:2503.19786, 2025. URL https://arxiv.org/abs/2503.19786.

Isaac Gibbs, John J. Cherian, and Emmanuel J. Candès. Conformal prediction with conditional guarantees. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 87(4): 1100–1126, 2025.

Yu Gu, Zijun Yu, Vahid Partovi Nia, and Masoud Asgharian. Pause and Reflect: Conformal Aggregation for Chain-of-Thought Reasoning. arXiv preprint arXiv:2605.14098, 2026. URL https://arxiv.org/abs/2605.14098.

Marc Hallin. Measure transportation and statistical decision theory. *Annual Review of Statistics and Its Application*, 9:401–424, 2022.

Marc Hallin and Dimitri Konen. Multivariate quantiles: Geometric and measure-transportation-based contours. In Vladik Kreinovich, Woraphon Yamaka, and Supanika Leurcharusmee (eds.), *Applications of Optimal Transport to Economics and Related Topics*, volume 556 of *Studies in Systems, Decision and Control*, pp. 61–78. Springer, Cham, 2024. doi: 10.1007/978-3-031-67770-0_6.

Jiuzhou Han, Wray Buntine, and Ehsan Shareghi. Towards Uncertainty-Aware Language Agent. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 6662–6685. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-acl.398. URL https://aclanthology.org/2024.findings-acl.398/.

Michal Klein, Louis Béthune, Eugène Ndiaye, and Marco Cuturi. Multivariate conformal prediction using optimal transport. *Transactions on Machine Learning Research*, 2026. URL https://openreview.net/forum?id=LrXAq63eT7.

Vladimir Kondratyev, Alexander Fishkov, Nikita Kotelevskii, Mahmoud Hegazy, Rémi Flamary, Maxim Panov, and Eric Moulines. Neural Optimal Transport Meets Multivariate Conformal Prediction. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=ylaKpd7tmA.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=VD-AYtP0dve.

Minjae Lee, Kyungmin Kim, Taesoo Kim, and Sangdon Park. Selective Generation for Controllable Language Models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://papers.nips.cc/paper/2024/hash/5a6815122f533193a022cbc41786c1cc-Abstract-Conference.html.

Jun Li, Juan A. Cuesta-Albertos, and Regina Y. Liu. DD-classifier: Nonparametric classification procedure based on DD-plot. *Journal of the American Statistical Association*, 107(498):737–753, 2012.

Moule Lin, Qizhen Lan, Shuhao Guan, Weipeng Jing, Jiexin Fan, David Gregg, and Goetz Botterweck. Uncertainty Quantification for LLM Agents: A Taxonomy, an Evaluation Protocol, and an Empirical Study. arXiv preprint arXiv:2609.07395, 2026. URL https://arxiv.org/abs/2609.07395.

Terrance Liu and Steven Wu. Multi-group Uncertainty Quantification for Long-form Text Generation. In *Proceedings of the Forty-first Conference on Uncertainty in Artificial Intelligence*, volume 286 of *Proceedings of Machine Learning Research*, pp. 2659–2684. PMLR, 2025. URL https://proceedings.mlr.press/v286/liu25a.html.

Xun Liu, Yi Evie Zhang, Vira Kasprova, Parisa Rabbani, Pardis Sadat Zahraei, Tianyu Zhang, Ali Ebrahimpour-Boroojeny, and Varun Chandrasekaran. AgentAbstain: Do LLM Agents Know When Not to Act? arXiv preprint arXiv:2607.10059, 2026. URL https://arxiv.org/abs/2607.10059.

Rui Luo and Zhixin Zhou. Conditional Conformal Risk Adaptation. arXiv preprint arXiv:2504.07611, 2025. URL https://arxiv.org/abs/2504.07611.

Andrey Malinin and Mark Gales. Uncertainty Estimation in Autoregressive Structured Prediction. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=jN5y-zb5Q7m.

Potsawee Manakul, Adian Liusie, and Mark Gales. SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9004–9017, 2023. doi: 10.18653/v1/2023.emnlp-main.557. URL https://aclanthology.org/2023.emnlp-main.557/.

Gregory Matsnev. Uncertainty Decomposition for Clarification Seeking in LLM Agents. arXiv preprint arXiv:2606.19559, 2026. URL https://arxiv.org/abs/2606.19559.

Soundouss Messoudi, Sébastien Destercke, and Sylvain Rousseau. Ellipsoidal conformal inference for multi-target regression. In *Proceedings of the Eleventh Symposium on Conformal and Probabilistic Prediction with Applications*, volume 179 of *Proceedings of Machine Learning Research*, pp. 294–306. PMLR, 2022. URL https://proceedings.mlr.press/v179/messoudi22a.html.

Meta Superintelligence Lab. Muse Glimmer 30B. Model card, https://huggingface.co/meta-models/Muse-Glimmer-30B, August 2026.

Mistral AI. Mistral Small 4. Model release, https://mistral.ai/news/mistral-small-4/, March 2026.

Christopher Mohri and Tatsunori Hashimoto. Language Models with Conformal Factuality Guarantees. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 36029–36047. PMLR, 2024. URL https://proceedings.mlr.press/v235/mohri24a.html.

Jyrki Möttönen and Hannu Oja. Multivariate spatial sign and rank methods. *Journal of Nonparametric Statistics*, 5(2):201–213, 1995.

Elizbar A. Nadaraya. On estimating regression. *Theory of Probability & Its Applications*, 9(1): 141–142, 1964.

Changdae Oh, Seongheon Park, To Eun Kim, Jiatong Li, Wendi Li, Samuel Yeh, Sean Du, Hamed Hassani, Paul Bogdan, Dawn Song, and Sharon Li. Uncertainty Quantification in LLM Agents: Foundations, Emerging Challenges, and Opportunities. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16219–16250. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.738. URL https://aclanthology.org/2026.acl-long.738/.

Jeffery Opoku and David Banahene. ToolChain-CRC: Conformal Risk Control for Agentic AI Under Retrieval and Tool-Use Drift. arXiv preprint arXiv:2606.18467, 2026. URL https://arxiv.org/abs/2606.18467.

Yash Patel, Eduardo Ochoa Rivera, and Ambuj Tewari. Conformal Prediction for Ensembles: Improving Efficiency via Score-Based Aggregation. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/fc1dce23e0ba01d3b691789cfd4f65c3-Paper-Conference.pdf.

Shishir G. Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. The Berkeley function calling leaderboard (BFCL): From tool use to agentic evaluation of large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48371–48392. PMLR, 2025. URL https://proceedings.mlr.press/v267/patil25a.html. Leaderboard https://gorilla.cs.berkeley.edu/leaderboard.html; package bfcl-eval, v3 multi-turn categories.

Dzianis Piatrashyn, Nikita Kotelevskii, Kirill Grishchenkov, Nikita Glazkov, Ivan Nasonov, Ilya Makarov, Timothy Baldwin, Preslav Nakov, Roman Vashurin, and Maxim Panov. ReDAct: Uncertainty-Aware Deferral for LLM Agents. arXiv preprint arXiv:2604.07036, 2026. URL https://arxiv.org/abs/2604.07036.

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S. Jaakkola, and Regina Barzilay. Conformal Language Modeling. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/31421b112e5f7faf4fc577b74e45dab2-Paper-Conference.pdf.

Md Ashikur Rahman, Md Arifur Rahman, Niamul Hassan Samin, Khandaker Rifah Tasnia, Md Hasibul Amin, Sifat Rahman Ahona, and Juena Ahmed Noshin. Beyond Aggregate Risk: Role-Stratified Conformal Risk Control for LLM Tool Calls. arXiv preprint arXiv:2607.24343, 2026. URL https://arxiv.org/abs/2607.24343.

Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, Zhenjia Xu, Dorsa Sadigh, Andy Zeng, and Anirudha Majumdar. Robots That Ask For Help: Uncertainty Alignment for Large Language Model Planners. In *Proceedings of the 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*. PMLR, 2023. URL https://proceedings.mlr.press/v229/ren23a.html.

Yaniv Romano, Rina Foygel Barber, Chiara Sabatti, and Emmanuel Candès. With malice toward none: Assessing uncertainty via equalized coverage. *Harvard Data Science Review*, 2(2), 2020.

Maxon Rubin-Toles, Maya Gambhir, Keshav Ramji, Aaron Roth, and Surbhi Goel. Conformal Language Model Reasoning with Coherent Factuality. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/679fcceef65c3d855aa885bd024542c1-Abstract-Conference.html.

Murilo Salem, Luísa Böhm, Daniel Pontes, and Anderson Ferrugem. Hierarchical Group-Conditional Conformal Risk Control for Selective Prediction in Language Models. arXiv preprint arXiv:2607.24562, 2026. URL https://arxiv.org/abs/2607.24562.

Andrea Santilli, Adam Golinski, Michael Kirchhof, Federico Danieli, Arno Blaas, Miao Xiong, Luca Zappella, and Sinead Williamson. Revisiting Uncertainty Quantification Evaluation in Language Models: Spurious Interactions with Response Length Bias Results. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 743–759, 2025. doi: 10.18653/v1/2025.acl-short.60. URL https://aclanthology.org/2025.acl-short.60/.

Carlo Sguera, Pedro Galeano, and Rosa E. Lillo. Spatial Depth-Based Classification for Functional Data. *TEST*, 23(4):725–750, 2014. doi: 10.1007/s11749-014-0379-1.

Wenwen Si, Sooyong Jang, Insup Lee, and Osbert Bastani. Conformal Constrained Policy Optimization for Cost-Effective LLM Agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 25446–25453, 2026. doi: 10.1609/aaai.v40i30.39739. URL https://ojs.aaai.org/index.php/AAAI/article/view/39739.

Sina Tayebati, Divake Kumar, Nastaran Darabi, Davide Ettori, Ranganath Krishnan, and Amit Ranjan Trivedi. TRACER: Trajectory Risk Aggregation for Critical Episodes in Agentic Reasoning. arXiv preprint arXiv:2602.11409, 2026. URL https://arxiv.org/abs/2602.11409.

Gauthier Thurin, Kimia Nadjahi, and Claire Boyer. Optimal Transport-based Conformal Prediction. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 59509–59527. PMLR, 2025. URL https://proceedings.mlr.press/v267/thurin25a.html.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher Manning. Just Ask for Calibration: Strategies for Eliciting Calibrated Confidence Scores from Language Models Fine-Tuned with Human Feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5433–5442, 2023. doi: 10.18653/v1/2023.emnlp-main.330. URL https://aclanthology.org/2023.emnlp-main.330/.

Ryan J. Tibshirani, Rina Foygel Barber, Emmanuel J. Candès, and Aaditya Ramdas. Conformal prediction under covariate shift. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.

Roman Vashurin, Maiya Goloburda, Preslav Nakov, and Maxim Panov. UNCERTAINTY-LINE: Length-Invariant Estimation of Uncertainty for Large Language Models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 7881–7908, 2025. doi: 10.18653/v1/2025.emnlp-main.400. URL https://aclanthology.org/2025.emnlp-main.400/.

Artem Vazhentsev, Akim Tsvigun, Roman Vashurin, Sergey Petrakov, Daniil Vasilev, Maxim Panov, Alexander Panchenko, and Artem Shelmanov. Efficient Out-of-Domain Detection for Sequence to Sequence Models. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 1430–1454, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.93. URL https://aclanthology.org/2023.findings-acl.93/.

Vladimir Vovk, David Lindsay, Ilia Nouretdinov, and Alex Gammerman. Mondrian confidence machine. Technical report, Royal Holloway, University of London, 2003.

Xi Wang, Anushri Suresh, Alvin Zhang, Rishi More, William Jurayj, Benjamin Van Durme, Mehrdad Farajtabar, Daniel Khashabi, and Eric Nalisnick. Conformal Thinking: Risk Control for Reasoning on a Compute Budget. In *Forty-third International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2602.03814.

Geoffrey S. Watson. Smooth regression analysis. *Sankhyā: The Indian Journal of Statistics, Series A*, 26(4):359–372, 1964.

Yunpeng Xu, Wenge Guo, and Zhi Wei. Selective Conformal Risk Control. arXiv preprint arXiv:2512.12844, 2025. URL https://arxiv.org/abs/2512.12844.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. WebShop: Towards Scalable Real-World Web Interaction with Grounded Language Agents. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/82ad13ec01f9fe44c01cb91814fd7b8c-Abstract-Conference.html.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Caiqi Zhang, Ruihan Yang, Xiaochen Zhu, Chengzu Li, Tiancheng Hu, Yijiang River Dong, Deqing Yang, and Nigel Collier. Confidence Estimation for LLMs in Multi-turn Interactions. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 25661–25676. Association for Computational Linguistics, 2026a. doi: 10.18653/v1/2026.findings-acl.1280. URL https://aclanthology.org/2026.findings-acl.1280/.

Jiaxin Zhang, Prafulla Kumar Choubey, Kung-Hsiang Huang, Caiming Xiong, and Chien-Sheng Wu. Agentic Uncertainty Quantification. arXiv preprint arXiv:2601.15703, 2026b. URL https://arxiv.org/abs/2601.15703.

Qiwei Zhao, Dong Li, Yanchi Liu, Wei Cheng, Yiyou Sun, Mika Oishi, Takao Osaki, Katsushi Matsuda, Huaxiu Yao, Chen Zhao, Haifeng Chen, and Xujiang Zhao. Uncertainty Propagation on LLM Agent. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6064–6073. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.302. URL https://aclanthology.org/2025.acl-long.302/.

Xiaofan Zhou, Huy Nguyen, Bo Yu, Chenxi Liu, and Lu Cheng. Adaptive Stopping for Multi-Turn LLM Reasoning. arXiv preprint arXiv:2604.01413, 2026. URL https://arxiv.org/abs/2604.01413.

A Related Work

Uncertainty signals in language models. Language-model uncertainty estimates differ in the information they use. Token-probability methods score a generated sequence from its log-likelihood (Malinin & Gales, 2021), including the negative log-likelihood of a single greedy generation (Aichberger et al., 2026). SAR adds semantic relevance weights at the token level and a sentence-level component across sampled generations (Duan et al., 2024). SelfCheckGPT compares sampled responses through similarity, entailment or prompting (Manakul et al., 2023), while semantic entropy groups generations by meaning (Kuhn et al., 2023; Farquhar et al., 2024). Verbalized confidence can accompany an answer or be elicited by a further prompt (Tian et al., 2023). Our primary step signal uses the geometric-mean token probability recorded in one rollout (Section 3). Length dependence arises at two scales. UNCERTAINTY-LINE adjusts generation-level scores by regressing out their dependence on output length (Vashurin et al., 2025), and length biases shared by the correctness function can distort the evaluation of uncertainty scores (Santilli et al., 2025). We examine the additional dependence introduced when per-step signals are combined over a variable number of agent steps.

Uncertainty along agent trajectories. Agent uncertainty methods account for how uncertainty evolves across decisions. SAUP weights step uncertainties by situational context (Zhao et al., 2025); UProp separates intrinsic uncertainty from uncertainty inherited from earlier decisions, using sampled decision processes (Duan et al., 2025). TRACER combines surprisal, situational awareness and tool-grounded coherence through a tail-focused aggregation that emphasizes critical episodes; its breakdown bound relies on structural assumptions on these step risks (Tayebati et al., 2026). Uncertainty also drives online interventions: UALA decides when to consult tools (Han et al., 2024), and ReDAct routes decisions to a larger model under an average deferral budget (Piatrashyn et al., 2026). Recent work studies confidence calibration as dialogue accumulates information (Zhang et al., 2026a), tensor representations of sampled multi-agent reasoning trajectories (Chen et al., 2026), and the relation between step-level and trajectory-level uncertainty (Oh et al., 2026; Lin et al., 2026). Comparative studies make the aggregation itself a central issue: Bouchard & Chauhan (2026) find consistency methods often strongest and token-probability scores sensitive to the turn aggregator, while Matsnev (2026) shows that length-only surrogates can match products of verbalized step confidences in failure-detection AUROC. We compare selected aggregation operators on a common recorded signal, with each adaptation documented in Table 3; for UALA, this is a terminal-step proxy. Holding the marginal risk cap fixed then exposes how score design affects both commitment and the distribution of accepted errors across trajectory lengths.

Conformal generation and selective release. Conformal abstention is the direct antecedent of our decision objective. Abbasi-Yadkori et al. (2024) calibrate release using similarity across sampled responses, and Gu et al. (2026) apply CRC to weighted aggregates of sampled reasoning paths. Both target the joint event that an answer is released and incorrect. Other conformal methods construct answer sets with coverage guarantees (Quach et al., 2024) or filter generated claims under statistical factuality guarantees (Mohri & Hashimoto, 2024), with extensions to learned scores and conditional validity (Cherian et al., 2024) and to deductively coherent reasoning (Rubin-Toles et al., 2025). Lee et al. (2024) control an entailment-based false discovery rate among released generations. The choice of risk distinguishes these guarantees. SCoRE controls either marginal deployment risk or a selective risk of false-discovery type among trusted outputs (Bai & Jin, 2026); selective CRC calibrates on a confidence-selected subset (Xu et al., 2025). Tail-risk control (Chen et al., 2025) and calibrated reasoning stops (Wang et al., 2026) further extend the objectives and decision times considered. We retain the marginal wrong-commit objective and compute the score from one recorded agent rollout. Under exchangeability, CRC bounds its expected loss over calibration and test draws, conditional on the independent reference data (Section 2.3). Error conditional on release is a separate quantity.

Conformal decisions for interactive agents. ToolChain-CRC is the closest trajectory-level antecedent: it treats a retrieval and tool-use run as one exchangeable unit and uses CRC to calibrate an accept-or-intervene rule, with extensions for drift and anytime monitoring (Opoku & Banahene, 2026). It aggregates stage-specific risk signals and calibrates against an audited trajectory loss covering retrieval support, tool validity and synthesis. Our label is terminal correctness, and we evaluate its noisy-OR aggregation on the same recorded token-probability signal as the other scores (Table 3).

Other methods intervene during the interaction. KnowNo calibrates action sets and requests human help when they are ambiguous, providing a statistical task-success guarantee under its assumptions on help and execution (Ren et al., 2023). Conformal information pursuit chooses queries by minimizing estimated expected prediction-set size (Chan et al., 2025); MiCP supports early stopping in multi-turn reasoning (Zhou et al., 2026); CCPO learns an orchestration policy under a conformal reliability constraint (Si et al., 2026). CORA applies CRC to an execute-or-abstain decision before each GUI action (Feng et al., 2026b), while conformal error attribution returns contiguous step sets with coverage guarantees for the decisive error (Feng et al., 2026a). Our gate governs the release of a completed rollout's output (Section 2.1). This distinction matters for irreversible tool actions, where abstention after execution can be too late (Liu et al., 2026).

Marginal risk and heterogeneous error allocation. Marginal risk control constrains a population average and leaves its allocation across groups open (Angelopoulos et al., 2024; Bates et al., 2021). Conformal Risk Adaptation is a close antecedent to our approach: Luo & Zhou (2025) adapt the score for image segmentation to make false-negative rates more consistent across images while preserving marginal validity; a stratified variant adds group-specific thresholds. We develop the score-design perspective for terminal agent release, using ordered uncertainty paths and evaluating the allocation of wrong commits across trajectory lengths. A complementary route specifies conditional requirements during calibration. Mondrian and equalized-coverage methods calibrate within declared groups (Vovk et al., 2003; Romano et al., 2020). Conditional conformal methods accommodate covariate function classes (Gibbs et al., 2025) and overlapping or fractional memberships (Bairaktari et al., 2025), while adaptive risk control learns threshold adjustments from data (Blot et al., 2025). For language models, multicalibration and multivalid conformal prediction address demographic subgroup reliability (Liu & Wu, 2025); recent preprints calibrate over group hierarchies (Salem et al., 2026) or semantic roles in tool calls (Rahman et al., 2026). These methods target specified conditioning structures. Our pooled comparison examines the empirical trade-off between commitment and error allocation under a common marginal cap (Section 4).

Geometry of trajectory scores. Spatial ranks average unit directions from a reference cloud (Möttönen & Oja, 1995); their relation to geometric quantiles provides a multivariate notion of order (Chaudhuri, 1996; Hallin & Konen, 2024). For a spatial-rank norm r, spatial depth is $1 - r$, so TSC contrasts class-wise depths in the spirit of depth–depth classification (Li et al., 2012). Dutta et al. (2016) learn classifiers from class-wise depths localized in feature space, and Sguera et al. (2014) classify curves from functional spatial depths, including a kernel-localized variant. Our score uses a fixed contrast, and its length-local variant L-TSC localizes in $\log T$ rather than in curve space, weighting reference paths with one prespecified bandwidth (Equation (6)). Multivariate conformal methods construct ellipsoidal regions (Messoudi et al., 2022) or use transport-based ranks (Chernozhukov et al., 2017; Hallin, 2022) and estimated transport maps (Thurin et al., 2025; Klein et al., 2026; Kondratyev et al., 2026). Our construction follows the conformal use of geometric spatial ranks by Conrad et al. (2026). Patel et al. (2025) also exploit joint score structure, constructing quantile envelopes over ensemble scores for prediction regions with coverage guarantees. Here, correctness-labeled reference paths determine a supervised score for terminal release under CRC. Ordered paths let this score use configurations of uncertainty across steps that symmetric and additive summaries can miss (Proposition A1). The resulting comparison connects trajectory representation to commitment and the allocation of wrong commits.

B Proofs and properties of the score

Reduction to conformal risk control.

Theorem A1 (Bound (9), restated)**.** *Under Assumption 1, for any $\alpha \in (0, 1)$ and any episode score s that is a fixed measurable function given the reference data, the rule* $\mathsf{commit}(\tau) = \mathbb{1}\{s(\tau) < \hat{\lambda}\}$ *with $\hat{\lambda}$ from Equation* (8) *satisfies* $\Pr\big(\mathsf{commit}(\tau_{n+1}) = 1 \wedge \hat{y}_{n+1} \neq y^{\star}_{n+1}\big) \leq \alpha$*, jointly over the calibration and test draws given the reference data.*

Proof. This is the conformal risk control theorem (Angelopoulos et al., 2024, Thm. 1) after two checks. (i) Given the reference data, every fitted component of s (the clouds and kernel lengths of Section 2.2, the (k, w) pair of the TRACER tail functional, the worst-step ECDF) is fixed, so the losses $L_i(\lambda)$

of Equation (7) are one fixed measurable functional of exchangeable tuples, hence exchangeable. (ii) In $\mu = -\lambda$, each L_i is non-increasing, right-continuous, $[0, 1]$-valued and zero at $\mu = +\infty$, the hypotheses of their theorem with $B = 1$, and their threshold $\hat{\mu} = \inf\{\mu : (n\tilde{R}_n(\mu)+1)/(n+1) \le \alpha\}$ is $-\hat{\lambda}$ for the supremum in Equation (8). Their conclusion $\mathbb{E}[L_{n+1}(\hat{\lambda})] \le \alpha$ is the displayed probability. □

Closed form of the threshold.

Theorem A2 (Order-statistic form, as implemented)**.** *Let $w_{(1)} \le \cdots \le w_{(K)}$ be the ordered scores of the wrongly answered calibration episodes and $q = \lfloor \alpha(n+1) \rfloor - 1$. Then Equation* (8) *equals $-\infty$ if $q < 0$; $w_{(q+1)}$ if $0 \le q < K$; and $+\infty$ if $q \ge K$.*

Proof. Let $N(\lambda) = n\widehat{R}_n(\lambda) = \#\{k : w_{(k)} < \lambda\}$, the number of wrong calibration scores strictly below λ. Since N is integer-valued, the constraint of Equation (8), $N(\lambda) \le \alpha(n+1) - 1$, reads $N(\lambda) \le q$. If $q < 0$ no λ is feasible and the supremum of the empty set is $-\infty$ (never commit). If $q \ge K$, every λ, $+\infty$ included, satisfies $N(\lambda) \le K \le q$, and the supremum is $+\infty$ (always commit). If $0 \le q < K$, then $N(\lambda) \le q$ for $\lambda \le w_{(q+1)}$ (at most $w_{(1)}, \ldots, w_{(q)}$ lie strictly below) and $N(\lambda) \ge q+1$ for $\lambda > w_{(q+1)}$; the feasible set is $(-\infty, w_{(q+1)}]$ and its supremum $w_{(q+1)}$ is attained. □

The strict commit inequality excludes the episodes tied at the selected threshold, so discrete or heavily tied scores can land at conservative empirical operating points (visible for the coordinatewise Bonferroni in Figure 3 and for the integer-valued length-only gate, Appendix C).

Properties of the spatial-rank contrast. Write $\mathrm{sgn}(u - v) = (u - v)/\|u - v\|$ for $v \ne u$ and $\mathrm{sgn}(0) = 0$. Then Equation (4) is the norm of the mean spatial sign over the cloud and Equation (5) the difference of two such norms. Three elementary properties are used later.

(i) Bounds and ties. The cloud C is any finite non-empty multiset. Repeated points contribute repeated unit vectors, and a query that coincides with m points of the cloud loses $m/|C|$ of the mass. Each summand has norm at most one, so by the triangle inequality $0 \le r(u; C) \le 1$ and $-1 \le s(u) \le 1$. The value $r = 1$ is approached when every $u - v$ points the same way, i.e. far outside the cloud.

(ii) One reference point versus the query. Each reference point contributes a vector of norm at most $1/|C|$, so replacing one reference point moves r by at most $2/|C|$. The query has no such bound near a tie: at $u = v$ the summand is 0, and at $u = v + \epsilon$ it is a unit vector for every $\epsilon \ne 0$. This discontinuity is what breaks the exact separation of Proposition A1 under perturbation (see the noise sweep below).

(iii) Invariances and the length kernel. r is invariant to translations, to positive rescalings of the whole space and to orthogonal maps. It is not invariant to general affine maps or to coordinate-wise rescalings, since the Euclidean norm fixes the metric; two whitened metrics are among the variants of Table 14. The length-kernel weights $w_j > 0$ of Section 2.2 are normalized within each cloud, so r_h is well defined and the bounds of (i) hold unchanged.

Representational separation. Symmetric aggregations (mean, max, RMS, pseudo-Noisy-OR) are permutation-invariant. Positional additive scores $\sum_d f_d(U_d)$ keep each position, but their class expectations are fixed by the coordinate marginals. The contrast Equation (5) is neither, and a fixed-length witness isolates the difference (Figure 2).

Proposition A1 (Separation beyond symmetric and additive scores)**.** *Let $a = \frac{1}{4}$, $b = \frac{1}{2}$, $c = \frac{3}{4}$, and let C^+ and C^- be the equal-weight clouds of the prototypes*

$$C^+ : (a, b, c),\ (b, c, a),\ (c, a, b), \qquad C^- : (a, c, b),\ (c, b, a),\ (b, a, c),$$

each completed by a constant terminal coordinate $\frac{1}{2}$. Then:

(i) *the two classes have identical coordinate marginals, hence identical mean vectors, and every permutation-invariant summary of the coordinates takes the same value on all six prototypes;*

(ii) *no score $\sum_d f_d(U_d)$ with real-valued f_d, nor a strictly increasing transformation of one, separates the two supports perfectly by a threshold;*

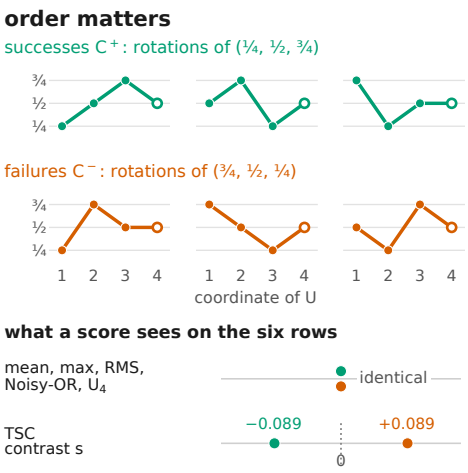

Figure 2: **The witness of Proposition A1.** Six length-four prototypes, three per class (hollow marker: the constant fourth coordinate). Every symmetric summary and the terminal coordinate take one value on all six; the contrast Equation (5) takes opposite signs. Exact on the ideal clouds only (noise sweep in Appendix B).

(iii) *evaluated at the prototypes themselves, with the zero-summand convention of Equation* (4)*, the contrast Equation* (5) *equals* $-(2-\sqrt{3})/3$ *on every success prototype and* $+(2-\sqrt{3})/3$ *on every failure prototype, so the threshold* 0 *separates them perfectly.*

Proof. Write the six prototypes as $(a_{\pi(1)}, a_{\pi(2)}, a_{\pi(3)}, \frac{1}{2})$ for the permutations π of $(a,b,c) = (\frac{1}{4}, \frac{1}{2}, \frac{3}{4})$: C^+ holds the three cyclic (even) permutations and C^- the three transpositions (odd ones), each with weight $\frac{1}{3}$.

(i) Over the three prototypes of either class each of the first three coordinates takes each of a, b, c exactly once, so the marginals are $\mathrm{Unif}\{a,b,c\}$ in both classes and the fourth coordinate is the constant $\frac{1}{2}$; the mean vectors coincide at $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})$. A permutation-invariant summary depends on the multiset of coordinates, $\{a, b, c, \frac{1}{2}\}$ for all six prototypes.

(ii) Let $S(u) = \sum_d f_d(u_d)$ with real-valued f_d. By (i), $\mathbb{E}[S \mid +] = \sum_d \mathbb{E}[f_d(U_d) \mid +] = \sum_d \mathbb{E}[f_d(U_d) \mid -] = \mathbb{E}[S \mid -]$, each term depending on the marginal of U_d only. A threshold separating the two supports would put S strictly below it on one support and at or above it on the other, hence, each class putting positive mass on each of its three prototypes, strictly order the two expectations: a contradiction. If g is strictly increasing and a threshold separates the supports through $g \circ S$, then $S(u) < S(v)$ for every accepted u and rejected v, and any t with $\max_{\text{acc}} S < t \le \min_{\text{rej}} S$ separates them through S itself, which was just excluded.

(iii) Take $u = (a, b, c, \frac{1}{2}) \in C^+$. The two other prototypes of C^+ give $u - v = (-\frac{1}{4}, -\frac{1}{4}, \frac{1}{2}, 0)$ and $(-\frac{1}{2}, \frac{1}{4}, \frac{1}{4}, 0)$, both of norm $\sqrt{3/8}$, so

$$\sum_{v \in C^+ \setminus \{u\}} \operatorname{sgn}(u - v) \;=\; \frac{(-\frac{3}{4}, 0, \frac{3}{4}, 0)}{\sqrt{3/8}}, \qquad \text{of norm } \sqrt{3},$$

and, the summand of u itself being zero, $r(u; C^+) = \sqrt{3}/3 = 1/\sqrt{3}$. The three prototypes of C^- give $u - v = (0, -\frac{1}{4}, \frac{1}{4}, 0)$, $(-\frac{1}{2}, 0, \frac{1}{2}, 0)$ and $(-\frac{1}{4}, \frac{1}{4}, 0, 0)$, so

$$\sum_{v \in C^-} \operatorname{sgn}(u - v) \;=\; \frac{(0,-1,1,0)+(-1,0,1,0)+(-1,1,0,0)}{\sqrt{2}} \;=\; (-\sqrt{2},\, 0,\, \sqrt{2},\, 0), \qquad \text{of norm } 2,$$

and $r(u; C^-) = 2/3$, whence $s(u) = 1/\sqrt{3} - 2/3 = -(2-\sqrt{3})/3$. The cyclic shifts of the first three coordinates map each cloud onto itself and preserve the Euclidean norm, so the value is the same on all of C^+; a transposition of two of the first three coordinates exchanges C^+ and C^-, which gives $+(2-\sqrt{3})/3$ on C^-. □

Consequences and limits of the construction. On this balanced population a gate reading a symmetric summary or the terminal coordinate sees the same value on both classes, so under any fixed rule or external randomization independent of the label $\Pr(\mathsf{commit} \wedge \mathrm{wrong}) = \frac{1}{2}\Pr(\mathsf{commit})$: a budget α caps its commit rate at 2α, whereas the sign gate $s < 0$ commits on the successful half with zero wrong-commit mass. The exact separation is an equality property of the prototype clouds: it rests on the zero summand of a query that coincides with a reference point (item (iii) above). Perturbing each coordinate by independent noise with full support gives both class densities full support, after which no score separates the two supports by a threshold; the contrast then ranks the classes only in a graded sense. The comparison holds for the ideal equal-weight clouds; Section 4 measures the capacity with a calibrated threshold on sampled references.

Risk under a shift of the group mixture. The bound of Theorem A1 is marginal over the source group mixture p. Corollary A1 bounds the risk of the same rule, calibrated on the source, when deployment draws groups from a nearby mixture q; re-calibrating under a known q is the weighted conformal setting (Tibshirani et al., 2019). The corollary covers an unknown shift of bounded size and controls no group separately.

Exchangeability alone does not imply it. Take $n = 9$ calibration episodes, $\alpha = 0.1$, every label wrong, and all source scores equal to one common $\mathrm{Unif}(0,1)$ draw V. The source tuples are exchangeable. The threshold of Theorem A2 has order index $\lfloor \alpha(n+1) \rfloor - 1 = 0$ and equals V, so the strict rule never commits on a source test point and its risk is 0. A target score V' independent of V with the same marginal law commits with probability $\frac{1}{2}$, and its risk is $\frac{1}{2}$. A single group produces this, so the per-group masses of the source say nothing about the target. The i.i.d. requirement below excludes the case.

Corollary A1 (Risk under a shift of the group mixture)**.** *All statements are conditional on the reference data. Assume that the calibration tuples and a source test tuple are i.i.d.; that $G(\tau) \in \{1, \dots, K\}$ is a group label fixed before evaluation or measurable from the reference data, with $p_g = \Pr(G(\tau_{n+1}) = g)$; and define, for $p_g > 0$, $r_g = \Pr\big(\mathsf{commit}(\tau_{n+1}) = 1 \wedge \hat{y}_{n+1} \neq y^\star_{n+1} \mid G(\tau_{n+1}) = g\big)$, the probability over the calibration draw and the independent source test tuple, with* commit *the rule of Theorem A1. Let a deployment tuple τ' be independent of the calibration draw, with $\Pr(G(\tau') = g) = q_g$, $q_g > 0$ only if $p_g > 0$, and, given $G(\tau') = g$, the joint law of $(\tau', \hat{y}', y^{\star\prime})$ equal to that of a source test tuple given $G(\tau_{n+1}) = g$. Then $R_q = \Pr(\mathsf{commit}(\tau') = 1 \wedge \hat{y}' \neq y^{\star\prime}) = \sum_g q_g r_g \leq \max_g r_g$, and for every $\rho \geq 0$ and every q with $\mathrm{TV}(q,p) \leq \rho$,*

$$R_q \;\leq\; \textstyle\sum_g p_g r_g + \rho\big(\max_g r_g - \min_g r_g\big) \;\leq\; \alpha + \rho\big(\max_g r_g - \min_g r_g\big),$$

the extrema over $\{g : p_g > 0\}$. If the spread $\max_g r_g - \min_g r_g$ is positive and $\rho \leq p_{g^\star}$ for some $g^\star \in \arg\min_g r_g$, the first inequality is an equality for some q with $\mathrm{TV}(q,p) = \rho$.

Proof. Given the calibration draw, commit is a fixed rule; the source test tuple and the deployment tuple are both independent of that draw and share, within each group, the same law, so conditioning on the group and averaging over the calibration draw gives $R_q = \sum_g q_g r_g$ (only groups with $q_g > 0$, hence $p_g > 0$, contribute); this is at most $\max_g r_g$ since q is a probability vector. Write $\delta_g = q_g - p_g$; then $\sum_g \delta_g = 0$, so the positive and negative parts have equal total, $\sum_g \delta_g^+ = \sum_g \delta_g^- = \mathrm{TV}(q,p) \leq \rho$, and

$$\begin{aligned} R_q - \textstyle\sum_g p_g r_g \;&=\; \textstyle\sum_g \delta_g^+ r_g - \sum_g \delta_g^- r_g \\ &\leq\; \mathrm{TV}(q,p)\big(\max_g r_g - \min_g r_g\big) \;\leq\; \rho\big(\max_g r_g - \min_g r_g\big) \end{aligned}$$

(each δ_g^+ multiplies an r_g at most the maximum, each δ_g^- one at least the minimum, and $\delta_g^+ > 0$ requires $q_g > 0$, hence $p_g > 0$). For the second inequality, $\sum_g p_g r_g = \Pr(\mathsf{commit}(\tau_{n+1}) = 1 \wedge \hat{y}_{n+1} \neq y^\star_{n+1})$ by the law of total probability over $G(\tau_{n+1})$, which Theorem A1 bounds by α (i.i.d. tuples are exchangeable). For attainment, take $g^+ \in \arg\max_g r_g$, distinct from $g^\star$ since the spread is positive, and $q = p + \rho(e_{g^+} - e_{g^\star})$, a probability vector because $\rho \leq p_{g^\star}$ and $p_{g^+} + \rho \leq p_{g^+} + p_{g^\star} \leq 1$, with $\mathrm{TV}(q,p) = \rho$ and $R_q = \sum_g p_g r_g + \rho(r_{g^+} - r_{g^\star})$. □

Use in the experiments. The shift column of Table 11 evaluates the identity $R_q = \sum_g q_g r_g$ at $\rho = 0.2$ with plug-in estimates of r_g (seed-mean group rates) on each population's length groups.

Appendix D checks the mass condition under which each cell is the attained worst case of Corollary A1, and Appendix E checks the identity seed by seed at the fixed calibrated threshold (Table 13). The metric H of Section 4 and the H columns of Tables 8 and 11 report $\max_g p_g r_g / \sum_h p_h r_h$, the largest share of the realized mass; the bound charges the spread $\max_g r_g - \min_g r_g$ instead, which H does not determine. The within-group laws are assumed invariant. A change of model, prompt or tool set may move them together with the mixture.

C Baseline Scores as Implemented

Each score of Table 3 is a scalar functional of the per-step uncertainty signal $\phi \in [0,1]^T$ of Equation (1), calibrated through Equation (8), which supplies the only guarantee; none claims a native one. Steps with no generated tokens do not occur in the aligned data; an episode with an empty signal would score NaN and be dropped from calibration. Table 3 gives, for each score, the implemented formula, what it reads from the frozen labeled reference partition, its tuning grid and objective, and the adaptation from the native method; the literature these methods come from is discussed in Appendix A.

Pseudo-Noisy-OR on the two representations. The score grid of Table 8 runs pseudo-Noisy-OR over the raw steps (Table 3); the representation study of Section 4 runs a second instance, "pseudo-Noisy-OR on U", whose product runs over the four interpolated coordinates $U_d \in [0,1]$ of Equation (3) with the same clipping and boundary treatment, so that it receives exactly the observations of the other scores. On coordinates below one the score is a strictly increasing transformation of the symmetric sum $\sum_d -\log(1-U_d)$, so it ranks as a symmetric additive score. The ϕ_t are not calibrated failure probabilities and steps are dependent, so the independent-failure reading is heuristic; CRC does not require it, since Equation (9) calibrates any fixed scalar functional whatever its interpretation.

Mean, max, coordinatewise Bonferroni, length-only gate. Mean and max of ϕ are the standard endpoints (max coincides with the tail functional at $w = 1$), oriented as in Section 2.3, so they commit on low average (respectively low worst-step) uncertainty. The *coordinatewise Bonferroni* interpolates to $D = 32$ coordinates, calibrates each $\Phi^\star[d]$ at level α/D via Equation (8), and commits if *any* coordinate clears its threshold: each coordinate is a fixed map of the episode, so Equation (9) caps its wrong-commit mass at α/D and a union bound over the D disjuncts caps the disjunction at α with no independence assumption. The bound is loose because neighbouring coordinates of one interpolated path are near-collinear, so the rule operates far below its budget and documents the cost of union-bound calibration (Figure 3). Its disjunctive decision is the sublevel set of $\min_d(\Phi^\star[d] - \hat{\lambda}_d)$ only once the D thresholds are calibrated, so there is no single pre-calibration scalar to rank. The *length-only gate* scores $s = \log T$, reads no signal and commits on short episodes under the CRC threshold: a fixed map to which Equation (9) applies verbatim, and the predictive control for the hypothesis that accumulating and extremal functionals act as length classifiers (Section 4). It is run on the nine populations under E1 with global calibration (readings: Section 4). Its score takes one value per length, and the strict commit rule excludes the whole block tied at the threshold, so the gate spends less than its budget. No tie-breaking device is used.

Scores of the representation study: common protocol. The six scores of Section 4 receive the interpolated vector $U = \Phi^\star(\tau) \in [0,1]^4$ of Equation (3) and nothing else. The five aggregations of the grid read on U (mean, max and RMS of U, the final coordinate U_D, which equals ϕ_T under Equation (3), and pseudo-Noisy-OR on U; RMS of U is SAUP's Eq. 1 with unit weights read on U) fit nothing and consume no reference label. TSC is the contrastive rank of Equation (5), identical to the score grid up to 10^{-12}; it consumes the reference labels through the clouds. Every score runs on the six MuSiQue and BFCL populations under E1, five reference draws and fifty rotations, and again on the three WebShop populations under the held-out protocol (50 development draws, one fixed test split; Appendix D).

TSC and L-TSC variants. The variant sweep of Figure 6 runs three TSC configurations (Equation (5) at $D \in \{3,4,5\}$) and three L-TSC configurations at $D = 4$ (bandwidths $h \in \{0.10, 0.15, 0.25\}$ on $\log T$); $D = 4$ and $h = 0.15$ are the configurations fixed in advance.

Table 3: **Specification and fidelity of the adapted scores.** Every score is a scalar functional of ϕ calibrated by Equation (8); "reference access" is what the score fits on the frozen labeled reference partition. The second line of each block states the adaptation from the native method.

Score	Implemented formula	Reference access	Tuning grid / objective	Origin
final-step score (UALA-style proxy)	$s = \phi_T$, the signal of the terminal (answer-emission) step.	none	none	Han et al. (2024)
Adaptation: proxy. The whole final step under the common estimator, scored offline on trajectories whose tool use already happened, under a CRC threshold over all calibration episodes. UALA-S scores the extracted answer span with one of five functionals of the softmax-renormalized answer-token probabilities, thresholds at a quantile over correctly answered calibration examples, and gates tool use and human deferral online; none of these elements is kept.				
TRACER tail functional (ϕ)	$K = \max(1, \lfloor kT \rfloor)$ and $s = (1-w)\,\frac{1}{K}\sum_{\phi_t \in \mathrm{top}_K(\phi)} \phi_t + w \max_t \phi_t$; $w = 1$ is the max score, and the mean is the endpoint $(k, w) = (1, 0)$, outside the grid.	tunes (k, w) on reference labels	$k \in \{0.1, 0.25, 0.5\}$, $w \in \{0, 0.25, 0.5, 0.75, 1\}$, maximizing the reference AUROC of s against episode failure, deterministic tie-break	Tayebati et al. (2026)
Adaptation: only the trajectory aggregation $\rho_{k,w}$ is reused, applied to ϕ in place of the MAX-composite step risk $r_t = \max\{U_t, \alpha D_{\mathrm{rep}}, \beta D_o^A, \gamma D_o^U\}$ (content-filtered surprisal composed with repetition and divergence detectors); the tuning is a custom reference-set protocol, not the paper's pairwise-logistic coarse-to-fine search. Coherence and ℓ_∞-Lipschitz stability of $\rho_{k,w}$ carry over to ϕ; TRACER's failure-control results concern r_t.				
SAUP aggregation (ϕ backbone), `RMS` / `pos.`	$s = \sqrt{\frac{1}{T}\sum_t (w_t \phi_t)^2}$ (their Eq. 1); `RMS`: $w_t = 1$; `pos.`: $w_t = t/T$.	none	none	Zhao et al. (2025)
Adaptation: Eq. 1 faithful on the shared ϕ backbone; `RMS` is the paper's own unweighted control and `pos.` our label-free ramp, the exact positional scheme not being published in closed form: for constant $\phi_t = c$ the score decreases from c toward $c/\sqrt{3}$ as T grows, so equal per-step uncertainty reads as safer on longer trajectories. Not reproduced: the $U^T + U^A$ step score, the undocumented "logical term for numerical stability", and SAUP-HMMD's situational weights, which need manually annotated situational labels that our setting does not provide.				
Bonferroni worst-step typicality	$s = -T\left(1 - \widehat{F}(\max_t \phi_t)\right)$, $\widehat{F}$ the ECDF of step values pooled from the *correct* reference trajectories; a worst step above the reference support scores 0, one below it $-T$, and a reference partition with no correct episode is replaced by the single pooled value 0.	correct-reference step ECDF	none	ours; contrast Ren et al. (2023)
Adaptation: our engineered union-bound-shaped comparator, *not* KnowNo, which builds per-step candidate-action prediction sets from multiple-choice confidences, calibrates whole scenarios at sequence level (min over steps, one quantile), certifies task completion under accurate human help, and motivates sequence-level calibration partly as avoiding union-bound conservatism. No native validity: pooled steps are not exchangeable units, longer reference episodes weigh more in $\widehat{F}$, the plain ECDF lacks the $+1$ conformal correction, and the adaptive T would need a stopping-time argument; the CRC wrapper supplies the only guarantee.				
pseudo-Noisy-OR / length-normalized twin	$s = 1 - \prod_t (1 - \phi_t)$ and $s = 1 - \left(\prod_t (1 - \phi_t)\right)^{1/T}$, in log space with ϕ_t clipped to $1 - 10^{-15}$, the product over every raw step.	none	none	Opoku & Banahene (2026); Zhang et al. (2026b); Matsnev (2026)
Adaptation: the unnormalized form is ToolChain-CRC's trajectory score, there over stage-specific step risks (one minus a retrieval-support confidence, a tool-error flag, a verifier score); on ϕ it is a ranking heuristic, not a probability, and grows mechanically with T. The length-normalized twin removes that drift for constant paths. Matsnev (2026) evaluates it on verbalized step confidences (after Zhang et al., 2026b) and finds product aggregation matched by length-only surrogates.				

D Experimental Details

This appendix section contains details on the experiment. First, the dataset characteristics, splitting strategy, correctness criteria, and agent generation process. Then, it covers the evaluation protocol, including the exact formula for the per-step uncertainty signals and the contract for the trajectory scores (the conditions under which they are compared). Next, we present additional experiments and controls. Finally, computational cost and reproducibility details.

D.1 Benchmarks and Data Generation

Common protocol. Each population consists of the trajectories generated by one model on one benchmark. The trajectories are generated once, then stored and reused across experiments. Generation uses vLLM with temperature 0.7 and top-p 1.0. A fixed seed derived from the benchmark identifier controls generation randomness. Within each benchmark, the three models use the same

prompts, tools, and evaluation procedure. Differences in how the models are run, including chat templates, token budgets, reasoning channels, and per-turn limits, are listed in Table 4.

We do not limit the number of steps an agent can take to complete a task, as this would alter the trajectory-length distribution under study; the one exception is Mistral Small 4 on BFCL, which keeps the benchmark's limit of 20 steps per turn. An episode ends when the agent provides its answer (`Finish` on MuSiQue, a closing text turn on BFCL, or `buy now` on WebShop) or when its context overflows. Episodes ending in context overflow are retained, with a reward of 0 on WebShop.

Correctness is evaluated using each benchmark's criterion. On MuSiQue, an answer is correct if its F1 score under the official evaluator is at least 0.5. BFCL uses its `multi_turn_checker`, with early-ended episodes padded with empty turns that fail the checker. On WebShop, success requires a reward of exactly one, a criterion fixed a priori and the same for all three models.

Within each population, episodes are divided into three groups based on trajectory length, using terciles computed without reference to correctness labels. On WebShop, the tercile boundaries are determined from the development set. Both the score grid and the representation study use the same generated trajectories.

Table 4: **The nine populations.** One frozen generator per row; benchmark headers give the shared task protocol, the harness column the model-specific accommodations. Partitions: reference/calibration/test sizes (Table 7). Accuracy and step counts over the whole pool (rotation populations) or the fixed test split (WebShop; strict success, reward $= 1$, fixed a priori for every generator). Groups: the population's length terciles in steps, computed from lengths alone.

Population, generator	Model-specific harness	ref/cal/test	Acc. (%)	T: median (range)	Groups (steps)
MuSiQue (2,417 answerable dev questions): ReAct with search/lookup tools over a BM25 index; step = one (thought, action, observation) cycle; ends at `Finish` or context overflow; label = answer F1 ≥ 0.5 (official evaluator)					
MuSiQue Gemma 3 12B (instruction-tuned, bf16)	256-token steps	725/846/846	33.1	4 (1–19)	1–3 / 4–5 / 6+
MuSiQue Mistral Small 4 (119B, NVFP4)	256-token steps	725/846/846	38.1	10 (1–2,015)	1–7 / 8–15 / 16+
MuSiQue Muse Glimmer 30B (NVFP4 build)	4,096-token steps; `Action` recovered from the reasoning channel when the visible text carries none	725/846/846	42.4	15 (2–167)	2–11 / 12–20 / 21+
BFCL v3 multi-turn (four categories $\times$ 200 episodes, pooled; `bfcl_eval` 2026.3.23): step = one assistant response (text or tool calls); a turn ends when the assistant answers in text; label = BFCL `multi_turn_checker`					
BFCL pooled Mistral Small 4 (119B, NVFP4)	native `tools` interface; BFCL's 20-step per-turn limit	240/280/280	39.3	12 (2–75)	2–10 / 11–14 / 15+
BFCL pooled Gemma 3 12B (instruction-tuned, bf16)	vendored Hermes-style tool template; no per-turn cap; 4,096-token steps	240/280/280	11.1	10 (2–736)	2–8 / 9–11 / 12+
BFCL pooled Muse Glimmer 30B (NVFP4 build)	native XML tool calls; reasoning channel archived and replayed; no per-turn cap	240/280/280	42.5	17 (2–177)	2–14 / 15–22 / 23+
WebShop (1,000 development + 500 test goals; seeded environment): step = one THOUGHT/ACTION reply (`webshop_react_v1`); ends at `buy now` or context overflow (reward 0); label = strict success (reward $= 1$), the benchmark's own criterion, the same for every generator (Appendix D)					
WebShop Gemma 3 12B (bf16)	256-token steps	300/700/500	20.8	4 (3–2,799)	3 / 4–5 / 6+
WebShop Mistral Small 4 (119B, NVFP4)	256-token steps	300/700/500	24.8	5 (3–6,285)	3 / 4–12 / 13+
WebShop Muse Glimmer 30B (NVFP4 build)	4,096-token steps; reasoning parser	300/700/500	46.6	9 (3–283)	3–6 / 7–17 / 18+

MuSiQue. The 2,417 answerable questions from the MuSiQue-Ans development split (Trivedi et al., 2022) are answered by a ReAct agent (Yao et al., 2023) with search and lookup tools over a BM25 index of the released paragraphs. One thought–action–observation cycle constitutes a step. We use an F1 threshold of 0.5 rather than exact match to allow for answer paraphrases. Raising the threshold to 0.7 changes $9.2/9.3/9.1\%$ of the labels, while lowering it to 0.3 changes $3.7/4.7/7.9\%$ (Gemma 3 12B / Mistral Small 4 / Muse Glimmer 30B).

Table 5: **Accuracy per length group.** Per population: accuracy (%), accuracy within each length group of Table 4 (short/mid/long, %) and median number of steps, over the whole pool (rotation populations) or the fixed test split (WebShop).

Population	Acc.	Group acc.	Med. T
MuSiQue (Gemma 3 12B)	33.1	46/31/12	4
MuSiQue (Mistral S4)	38.1	55/32/24	10
BFCL pooled (Mistral S4)	39.3	44/44/28	12
BFCL pooled (Gemma 3 12B)	11.1	12/11/10	10
BFCL pooled (Muse Glimmer 30B)	42.5	65/43/18	17
MuSiQue (Muse Glimmer 30B)	42.4	53/43/32	15
WebShop (Gemma 3 12B)	20.8	19/37/10	4
WebShop (Mistral S4)	24.8	25/38/9	5
WebShop (Muse Glimmer 30B)	46.6	51/56/33	9

Muse Glimmer 30B generates reasoning in a separate channel, with a limit of 4,096 tokens per step. When the visible text contains no `Action` line, we recover it from the reasoning channel (66.4% of steps). Nevertheless, 11.6% of its steps yield no valid action and continue with an error observation.

Accuracy decreases across length terciles for all three models ($46/31/12\%$, $55/32/24\%$, and $52.6/42.6/31.6\%$, respectively). Mistral Small 4 generates particularly long trajectories (mean 35.8 steps, median 10). Its 131 episodes exceeding 100 steps (5.4% of the population, accuracy 8.4%) repeatedly execute earlier actions or fail to produce a valid action; 93 end at context overflow. These episodes are retained because we do not limit the number of steps per episode.

BFCL. We use the four v3 multi-turn categories (`base`, `miss_func`, `miss_param`, `long_context`), with 200 episodes each. The agents are evaluated using the official `bfcl_eval` package (2026.3.23). One assistant response constitutes a step, and tool calls are executed in the live environment. Malformed tool-call JSON produces a no-op call with an error observation. In `miss_func`, held-out functions are restored at the flagged turn using BFCL's own prompt. The four categories are combined before splitting the episodes into reference, calibration, and test sets.

Mistral Small 4 uses vLLM's native `tools` interface and BFCL's limit of 20 steps per turn. The other two models require changes to the official setup. Gemma 3 12B has no tool-calling template and uses a vendored Hermes-style template that groups consecutive tool results into one user turn. Muse Glimmer 30B uses its own serving image, which supports tool calls and stores reasoning in a separate field. The reasoning is replayed within a turn but never merged into the visible step. Neither model has a limit on the number of steps per turn, and no turn was forcibly terminated over the 800 episodes.

Accuracy in the four categories (base / long-context / missing-parameter / missing-function) is $50.0/47.5/34.0/25.5\%$ for Mistral Small 4, $20.0/13.0/10.0/1.5\%$ for Gemma 3 12B, and $50.0/46.0/35.0/39.0\%$ for Muse Glimmer 30B. Across length terciles, accuracy varies little for Gemma 3 12B ($11.9/11.2/10.2\%$) but decreases for Muse Glimmer 30B ($64.8/42.8/17.6\%$).

WebShop. WebShop (Yao et al., 2022) is a simulated e-commerce site with 1.18 million products and human-written instructions. The agent uses `search` and `click` actions to find a product and complete a purchase, receiving a reward in $[0, 1]$. The environment is seeded (233) before goals and prices are generated, so all three models receive the same instructions. We use 1,000 development goals for reference and calibration and 500 test goals for evaluation. All models use the same ReAct prompt. An episode ends when the agent selects `buy now` or reaches the context limit of 131,072 tokens.

An episode is considered correct if its reward is exactly one, the benchmark's own success criterion, fixed a priori and the same for all three models; labels are computed from the stored rewards. On the test set, accuracy is 20.8% for Gemma 3 12B, 24.8% for Mistral Small 4, and 46.6% for Muse Glimmer 30B. The first two fall below the $[0.25, 0.65]$ accuracy range targeted during the generation pilot. This range was not required by CRC, and we retain all three populations.

The generation pilot also checks the proportion of valid parsed actions (≥ 0.90), the variation in trajectory length, and the generation speed. Gemma 3 12B fails the parsing and length checks, with half of its development episodes ending after the minimum of three steps. Mistral Small 4 repeatedly produces replies that cannot be parsed. The environment returns an error after each such reply until the context overflows; 95% of its test steps are unparsed. We retain these two populations without filtering or regeneration.

On the test set, 48.8% of Gemma 3 12B episodes and 32.4% of Mistral Small 4 episodes have the minimum length. The length-only control cannot distinguish episodes within these groups and commits nothing at $\alpha \leq 0.20$ in either population. Category groups are defined by the goal's top-level product category (five categories), while length groups use the development-set terciles of Table 4.

D.2 Evaluation Protocol

Step signals. For each generated token j, the serving engine records the log-probability ℓ_j of the generated token and the top-20 token log-probabilities, from which we compute the signals of Table 6. All signals are oriented so that higher values indicate greater uncertainty. The final step is included in every trajectory score. Token entropy is estimated from the top-20 probabilities, giving a lower bound on full-vocabulary Shannon entropy. The main experiments use E1; E3 repeats the full grid as a sensitivity check (Table 15). The representation study uses E1 only, interpolated to the $D = 4$ grid of Equation (3).

Table 6: **Per-step signals.** J tokens per step; ℓ_j and m_j the chosen-token and position-argmax log-probabilities; $\mathcal{H}_j$ the entropy of the top-20 tokens; V the vocabulary size (262,144 Gemma 3, 131,072 Mistral, 202,048 Muse Glimmer). E1 is the primary signal and E3 the sensitivity signal (Table 15).

	Estimator	Inputs requirements	Role
E1	$1 - \exp(\frac{1}{J}\sum_j \ell_j)$	ℓ_j	primary: one minus geometric-mean of generated token probability; Normalized Sequence Probability, the length-normalized form of MSP (Vazhentsev et al., 2023)
E2	$1 - \exp(\frac{1}{J}\sum_j m_j)$	m_j	not reported: one minus the argmax-token geometric mean along the generated path; per-step, length-normalized analogue of G-NLL (Aichberger et al., 2026), which is defined on the greedy sequence
E3	$\frac{1}{J}\sum_j \mathcal{H}_j / \ln V$	top-20 token	sensitivity: Mean Token Entropy; lower bound on the full-vocabulary value
E4	$1 - \exp(\min_j \ell_j)$	ℓ_j	not reported: one minus the probability of the step's least likely generated token
E5	$1 - \exp(\sum_j \ell_j)$	ℓ_j	not reported: one minus total step probability, unnormalized by length; Maximum Sequence Probability (Fadeeva et al., 2023)
TokenSAR	$1 - \exp(-\mathrm{TokenSAR}(t))$	ℓ_j, one auxiliary relevance scorer	not reported: relevance-weighted token log-likelihoods (Duan et al., 2024)

Splits. We use two splitting procedures, summarized in Table 7: *rotation* and *held-out*. On MuSiQue and BFCL, for *rotation*, we fix the reference set and randomly divide the remaining episodes into calibration and test sets 50 times. Each method therefore uses the same reference set across these repetitions. We report 95% bootstrap intervals for paired differences in commit rate, using 2,000 resamples of the 50 repetitions.

On WebShop, the 500 *held-out* test episodes remain fixed. For each of 50 repetitions, we draw 300 reference episodes from the 1,000 development episodes and use the remaining 700 for calibration. The development and test sets come from one seeded shuffle of the benchmark's instruction list: positions 500–1,499 form the development set, and positions 0–499 form the test set. Each goal is run once by the same agent. Thus, the reported variation across repetitions reflects changes in the reference and calibration sets, not changes in the test tasks.

Length terciles on MuSiQue and BFCL are computed from the full set of episodes, without using correctness labels. Computing them from the reference set instead gives the same boundaries on four populations and changes one boundary by one step on the two Muse Glimmer populations. WebShop uses the development set to define its terciles, which need not divide the test set equally.

Every length group has population mass greater than the total-variation radius $\rho = 0.2$ used in Table 11. The condition of Corollary A1 is therefore met on all nine populations, and the reported shift values are attained maxima for the estimated group rates rather than loose upper bounds.

The CRC guarantee applies under the exchangeability assumption of Assumption 1. It concerns risk averaged over calibration and test draws, not the error rate observed on every individual test set. On WebShop, the observed wrong-commit rate exceeds the nominal limit in some experiments. These results are reported in Appendix E and Tables 8 and 15.

Table 7: **The two split protocols.** Partition sizes are in Table 4; random generators are seeded by the root seed and the seed index.

	Rotation (six MuSiQue and BFCL populations)	**Held-out** (three WebShop populations)
Reference	frozen 30% partition, a function of the root seed only	300 episodes redrawn at every seed from the 1,000-episode development pool
Calibration	half of the remainder, redrawn at every seed	the remaining 700 development episodes
Test	the other half	the benchmark's 500-episode split, fixed
Repetitions	50 rotations per reference draw	50 reference/calibration draws on one test set
Length groups	terciles of the full pool	33rd and 67th percentiles of the development pool

Trajectory score contract. Each method produces a scalar nonconformity score s, and the agent commits when $s(\tau) < \lambda$, using the threshold obtained via CRC of Equation (8) on the calibration set. The trajectory score depends on the episode and, where applicable, the reference set, but not on α. Coordinatewise Bonferroni is an exception: it uses D thresholds at α/D and a disjunctive commit rule (union of rules over the coordinates). It therefore has no scalar score.

D.3 Additional Experiments and Controls

Controls over the reference split. We use additional experiments to examine how the results depend on the reference set, the number of labeled episodes, the order of steps, and the choice of distance metric. Their results are reported in Appendix E.

Fresh reference draws. We repeat the experiments with four additional reference sets (root seeds 1–4). Each new reference set is used to fit the score components, including the reference clouds, the (k, w) parameters, and the worst-step ECDF. Together with the original reference set, this gives five reference draws and 50 rotations per draw on each MuSiQue and BFCL population. On WebShop, the reference set is already redrawn at every repetition. We repeat the 50 draws for each additional root seed, keeping the same test set. The resulting variation does not represent independent samples of test tasks.

Equal label budget. Methods that do not fit a score component on the reference set are calibrated using the combined reference and calibration sets. This gives $1{,}571$ labeled episodes on MuSiQue, 520 on BFCL, and $1{,}000$ on WebShop. Methods that fit a score component on the reference set keep their original partitions. All methods thus use the same number of labeled episodes and are evaluated on the same test sets. The validity of calibration on the combined set depends on the reference-draw procedure; it is not covered directly by the conditional guarantee for a fixed fitted score.

Step order and distance metric. To examine the role of step order, we shuffle steps $1, \ldots, T-1$ within each episode, leaving the final step unchanged. The permutation is determined by a hash of the episode's uncertainty values, so the same episode is shuffled identically across partitions and repetitions.

We also compare two changes to the distance metric. The first standardizes each coordinate using its reference-set standard deviation. The second whitens the coordinates using the inverse square root of the reference covariance, shrunk toward its scaled identity with weight 0.05, fixed in advance. Both transformations are determined from the reference set and evaluated on all nine populations under E1 with 50 seeds.

Finally, we vary the number of labeled reference episodes from 25 to the full reference set, using the same partitions (Appendix E and Figure 4).

Geometry variants of the representation. The sensitivity table of Appendix E (Table 14) changes one factor of the representation $U = \Phi^\star(\tau)$ at a time, at $D = 4$, against the two pre-specified configurations; every quantity a factor estimates is fitted on the reference partition alone, both labels pooled, and then frozen, so each variant is a fixed map given the reference partition. *Projection basis*: the interpolant is sampled at 32 points and represented by its D least-squares coefficients in a fixed basis (clamped cubic B-splines on a uniform knot vector, the cubic Bernstein basis at $D = 4$; shifted Legendre polynomials of degrees 0 to $D - 1$), or by its projections on the top-D principal directions of the centered reference samples (FPCA, unscaled). *Represented function*: the increments of the interpolant sampled at $D + 1$ points in place of its levels, or both blocks concatenated, the latter admitted only under full whitening, which sets the relative scale of the two blocks from the reference covariance instead of a hand-set weight. *Value transform*: each step value replaced by its value under the right-continuous ECDF of all reference steps before interpolation (PIT of the steps), or each interpolated coordinate replaced by its value under the reference ECDF of that coordinate (PIT per coordinate). *Time warp*: the D sampling points moved toward the episode end, $u = \log(1 + \gamma v)/\log(1 + \gamma)$ (log warp) or $u = 1 - (1 - v)^\gamma$ (tail warp) for v on the uniform grid, $\gamma = 2$. *Metric*: the diagonal and full whitening of the metric control above. The variants were run on the full grid after the parents were fixed, are not pre-registered, and none is selected.

Representation study. The study of Section 4 reuses the six pools and partitions above without change: five reference draws (root seeds 0–4, the pre-specified draw included) $\times$ 50 rotations, E1 in the uncertainty orientation, $U = \Phi^\star(\tau) \in [0, 1]^4$ on the $D = 4$ grid, each score fitted, if at all, on the reference partition alone, one global CRC threshold per score at $\alpha = 0.10$. A dispersion is the std across the five reference-draw means, and paired differences are taken at matched (draw, rotation) cells. In Figure 1, W and H are computed on each test split from its length groups and then averaged like the other metrics, whereas Table 2 computes them from group masses averaged over splits (Table 8); the two conventions give different levels, so only differences within one convention are compared. The bootstrap intervals over the 50 rotations of the pre-specified draw are intervals for that draw alone.

The WebShop extension came last, under its own protocol frozen before any of its results was computed. It runs the same scores on the three held-out pools (300 reference and 700 calibration episodes redrawn on each of 50 development draws, 500 fixed test episodes). There a level is mean $\pm$ std over the 50 draws, which never resample the test tasks.

D.4 Reproducibility and computational cost

Provenance and software. Every result row records the generation-cache hash, the signal-cache hash, the config checksum, the git commit and the library versions (Python 3.12, NumPy 1.26.4, SciPy 1.17.1 for the score grid; Python 3.13, NumPy 2.2, SciPy 1.15 and pandas 2.2 for the representation study). A per-population provenance manifest in the supplementary material lists the nine populations' cache hashes, partition sizes, label conventions and software versions. All six MuSiQue and BFCL populations were generated on a single NVIDIA DGX Spark (one GB10 superchip, 128 GB unified memory, vLLM server), including Mistral Small 4 (119B) served without tensor parallelism; only the WebShop generations used rented A100 and H100 hosts, each population with one frozen serving configuration. Calibration and evaluation are CPU-only. The representation study is a self-contained module in the supplementary material (reproduction commands and a self-test of its

TSC implementation against the repository one); its tables and figures are emitted by one script, which asserts the headline numbers in the text against the archived CSV outputs before writing.

Compute cost. The certificate is a negligible addition to the cost of running the agent, reference fit included. Every grid score was timed on one CPU thread of the DGX Spark (BLAS pinned to one thread, one untimed warm-up, median over the 50 splits; the machine was shared with unrelated services during the measurement, so differences under a factor of two are not read). The per-score, per-population timings are not printed here; the supplementary material holds them. Fitting TSC on the reference partition, which amounts to storing the two labeled clouds, takes 2.3 to 7.5 ms per population, six to 23 times less than the (k, w) grid search of the TRACER tail functional (14 to 127 ms). Scoring one episode takes 41 to 68 μs, one pass over the $|C^+| + |C^-|$ reference points (0.08 to 0.19 μs per point, the $O(n_{\mathrm{ref}} D)$ cost of Section 2). The whole certificate of one split, from the reference fit through the CRC threshold to the scoring of every test episode, takes at most 0.13 s. The scalar aggregations score an episode in 2.0 to 5.6 μs on eight populations, 16 to 34 times faster than the rank, and in 9.9 to 24 μs on WebShop with Mistral Small 4, whose episodes average 568 steps (a factor of five). At either scale the difference is invisible next to generating the episode, each step of which is at least one call to the generator; these timings exclude generation and step scoring (GPU).

E Additional Results

This appendix holds the evidence behind Section 4 in eight blocks, each on the nine populations. Two sampling protocols appear and their episodes and replicates are never pooled: calibration/test rotations at a frozen reference draw on the six MuSiQue and BFCL populations, and reference/calibration redraws on a fixed test split for the three WebShop populations, under a correctness convention fixed a priori (Appendix D). A nine-population summary is an unweighted average or count of population-level results with no common uncertainty estimate.

Validity and realized risk. Every non-degenerate score realizes its cap on the six rotation populations, while the fixed WebShop test split exposes the development-to-test excess that no rotation within a single pool can reveal (Figure 3). Equation (9) bounds the marginal wrong-commit mass in expectation over the calibration and test draws, under exchangeability and per pre-specified score, so a realized CER is a diagnostic of the bound, not a test of it. At $\alpha = 0.10$ the six rotation populations realize the cap to within half a point under both signals (CER columns of Tables 8 and 15). On WebShop each CER is one draw of that expectation: most scalars leave budget unspent, no cell exceeds 10.2 under E1, and the largest cell under E3 is 11.3 (max step score with Gemma 3).

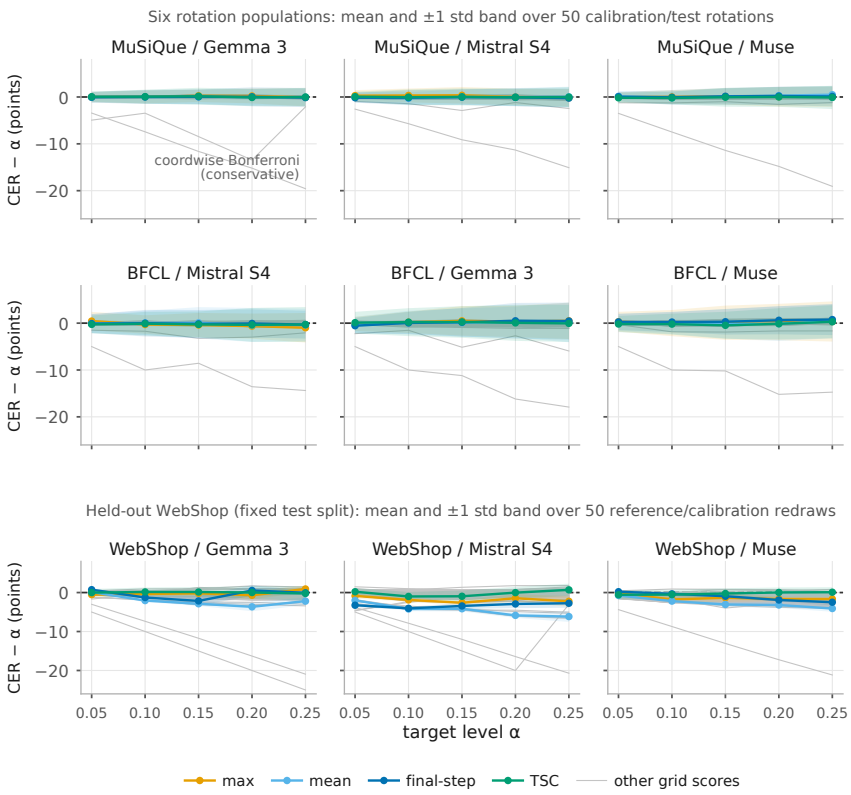

Figure 3: **Realized CER minus the target, $\mathrm{CER} - \alpha$, for every grid score (E1 signal).** Nine panels: the six rotation populations (top two rows) and the three held-out WebShop populations (bottom row), against the target $\alpha \in \{0.05, \ldots, 0.25\}$. The horizontal zero line is the cap; curves below it are conservative, visibly so for the coordinatewise Bonferroni, whose α/D union bound (and, at tight α on the pooled BFCL populations, the small-n feasibility limit of Equation (8)) leaves budget unspent. Highlighted scores carry ± 1 std bands, thin gray lines are the remaining grid. On the rotation populations the mean and band are over the 50 calibration/test rotations of the pre-specified reference draw; on WebShop they are over the 50 reference/calibration redraws on the one fixed test set, so each realized CER is one draw of the expectation bound and exceedances occur (readings at $\alpha = 0.10$: text of Appendix E). Equation (9) bounds an expectation, so a mean may land on either side of zero: the plot is a diagnostic of the realized risk, and an empirical exceedance does not by itself contradict the marginal expectation theorem.

Controls over the reference split. Table 8 carries the twelve-score grid behind the family-based subset of Table 2. The commit-rate leads of L-TSC on WebShop with Gemma 3 and with Mistral Small 4 are small, but their 95% redraw intervals on the fixed test split exclude zero. The interval of the difference between TSC and pseudo-Noisy-OR with Mistral Small 4 does not exclude zero.

Two controls qualify the leaders of Table 8. Across five independent reference draws (protocols: Appendix D) the MuSiQue leaders keep their places. On WebShop the reference is redrawn at every seed, so this control measures only the Monte-Carlo error of a 50-draw mean, a fraction of a point. Under an equal label budget, the scores with no fitted component also calibrate on the reference episodes. The leader is then unchanged on most populations and no fixed aggregation moves by more than a few points.

Table 8: **Full twelve-score grid on the nine populations ($\alpha = 0.10$, E1).** Per population: commit rate CR, realized wrong-commit rate CER, worst within-group CER W and largest single-group share of the wrong-commit mass H (%), over the population's length groups. CR: mean $\pm$ std over 50 seeds; W, H: $\pm$ half-width of a 95% paired bootstrap interval over the seeds. Best **bold**, second underlined within a population; the TRACER tail functional is unmarked where it coincides with max. †pre-specified primary method. Protocols: Appendix D.

MuSiQue	**Gemma 3 12B**				**Mistral S4**				**Muse Glimmer 30B**			
Score	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$
mean step score	24.8 ± 2.0	10.0	16.0±0.6	70±1	18.3 ± 2.0	9.7	24.0±0.8	80±1	14.4 ± 1.7	9.8	21.2±0.7	74±1
max step score	22.7 ± 2.2	10.0	19.6±0.7	85±1	26.8 ± 2.0	10.3	14.7±0.5	51±1	16.8 ± 1.6	10.0	11.8±0.5	39±1
pseudo-Noisy-OR	23.8 ± 1.9	10.0	21.5±0.8	94±1	**28.8 ± 2.1**	10.0	26.4±0.7	95±1	**20.6 ± 2.1**	9.9	28.5±0.9	95±1
pseudo-Noisy-OR, len-norm.	24.7 ± 1.9	9.9	16.1±0.6	70±1	18.5 ± 1.8	9.8	23.7±0.7	79±1	14.4 ± 1.8	9.8	21.4±0.7	75±1
SAUP (ϕ), RMS	24.2 ± 2.1	10.0	16.9±0.6	73±1	18.8 ± 1.9	9.8	23.2±0.7	77±1	14.4 ± 1.9	10.0	21.4±0.7	74±1
SAUP (ϕ), pos.	25.2 ± 2.0	10.1	15.3±0.5	66±1	18.3 ± 1.9	9.8	25.6±0.8	84±1	12.7 ± 1.9	9.9	23.1±0.9	80±1
TRACER tail (ϕ)	22.7 ± 2.2	10.0	19.6±0.7	85±1	26.8 ± 2.0	10.3	14.7±0.5	51±1	16.8 ± 1.6	10.0	11.8±0.5	39±1
Bonferroni worst-step	22.5 ± 2.2	10.0	14.2±0.7	62±1	18.8 ± 1.8	10.1	22.4±0.6	72±1	14.4 ± 1.9	10.3	22.0±0.7	74±1
coordwise Bonferroni	5.9 ± 1.3	2.6	4.1±0.3	42±2	7.5 ± 1.2	4.3	8.7±0.5	66±2	3.5 ± 0.9	2.5	4.9±0.5	66±3
final-step (UALA-style)	**26.8 ± 2.2**	10.0	12.5±0.6	54±1	21.0 ± 2.6	9.8	19.5±0.7	64±1	13.3 ± 2.0	9.9	18.9±0.7	65±1
TSC (D=4)†	23.7 ± 2.6	10.1	17.7±0.6	77±1	23.3 ± 2.5	10.0	15.0±0.8	49±1	17.8 ± 1.9	9.8	**11.4±0.5**	**38±1**
L-TSC (h=0.15)	25.0 ± 2.2	10.3	**11.7±0.7**	**48±1**	22.8 ± 2.8	9.8	**10.2±0.6**	**34±1**	19.9 ± 2.4	9.8	12.1±0.5	42±1

BFCL pooled	**Mistral S4**				**Gemma 3 12B**				**Muse Glimmer 30B**			
Score	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$
mean step score	18.6 ± 3.8	10.1	12.0±1.0	44±2	12.1 ± 2.9	10.1	14.1±1.1	45±2	22.6 ± 4.2	10.1	19.9±1.4	58±2
max step score	19.0 ± 2.8	9.7	15.6±0.9	62±2	**13.0 ± 2.5**	10.2	12.4±1.0	48±2	18.5 ± 3.9	10.0	**11.2±1.2**	**34±1**
pseudo-Noisy-OR	18.0 ± 3.7	10.0	23.8±1.6	91±1	12.9 ± 2.7	10.3	19.5±1.3	75±2	**30.5 ± 4.3**	9.4	23.8±1.5	84±2
pseudo-Noisy-OR, len-norm.	18.5 ± 4.0	10.1	12.1±1.0	45±2	12.3 ± 3.0	10.1	13.3±1.0	42±2	22.5 ± 4.3	10.0	19.6±1.5	57±2
SAUP (ϕ), RMS	18.9 ± 4.4	10.2	12.1±1.1	43±2	12.8 ± 2.8	9.9	**10.9±0.7**	43±2	21.8 ± 4.0	9.9	18.8±1.5	55±2
SAUP (ϕ), pos.	19.7 ± 4.1	10.5	13.0±1.3	39±2	12.7 ± 2.9	9.9	11.1±0.9	44±2	23.7 ± 3.6	10.1	22.6±1.3	66±2
TRACER tail (ϕ)	19.1 ± 2.9	9.7	15.5±0.9	61±2	12.7 ± 3.1	9.9	11.2±0.8	44±2	22.8 ± 3.9	10.2	19.3±1.5	55±2
Bonferroni worst-step	17.9 ± 3.3	9.8	11.3±0.9	**35±1**	12.2 ± 2.5	10.1	18.5±1.0	59±2	13.0 ± 4.2	10.5	28.9±2.0	81±2
coordwise Bonferroni	0.0 ± 0.0	0.0	0.0±0.0	—	0.0 ± 0.0	0.0	0.0±0.0	—	0.0 ± 0.0	0.0	0.0±0.0	—
final-step (UALA-style)	20.3 ± 3.9	10.0	**11.2±1.2**	39±2	10.4 ± 2.5	10.1	14.1±0.9	55±2	20.6 ± 3.0	10.2	22.6±1.5	64±2
TSC (D=4)†	**20.9 ± 3.2**	9.9	12.1±1.2	40±2	12.7 ± 3.3	10.2	11.3±1.0	**36±2**	21.7 ± 3.3	9.8	16.1±1.4	48±2
L-TSC (h=0.15)	20.1 ± 3.5	10.2	13.2±0.9	50±2	11.5 ± 2.9	10.1	12.4±1.1	48±2	19.3 ± 3.0	10.2	16.9±0.9	48±2

WebShop (held-out test)	**Gemma 3 12B**				**Mistral S4**				**Muse Glimmer 30B**			
Score	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$
mean step score	8.3 ± 0.4	7.9	25.6±0.1	89±1	6.0 ± 0.6	5.8	19.0±0.5	100±0	11.6 ± 0.4	7.8	18.1±0.1	74±0
max step score	13.5 ± 0.8	9.6	17.1±0.3	87±0	13.5 ± 0.5	8.1	20.0±0.2	80±0	19.3 ± 1.5	8.2	16.5±0.4	72±0
pseudo-Noisy-OR	13.0 ± 1.0	9.7	20.0±0.5	100±0	14.2 ± 1.1	10.1	30.4±0.8	98±0	**23.3 ± 1.7**	10.0	27.4±0.7	98±0
pseudo-Noisy-OR, len-norm.	8.4 ± 0.3	8.0	25.4±0.0	88±1	6.0 ± 0.5	5.9	19.6±0.4	100±0	12.2 ± 1.0	7.9	17.9±0.2	73±1
SAUP (ϕ), RMS	9.3 ± 0.9	8.6	24.3±0.1	78±1	6.6 ± 0.6	6.6	21.6±0.5	100±0	13.0 ± 0.8	8.4	16.6±0.1	63±1
SAUP (ϕ), pos.	8.8 ± 0.9	8.7	31.3±0.7	99±0	6.2 ± 0.4	6.2	20.3±0.4	100±0	11.5 ± 1.1	7.3	19.6±0.4	86±0
TRACER tail (ϕ)	13.5 ± 0.8	9.6	17.1±0.3	87±0	13.5 ± 0.5	8.1	20.0±0.2	80±0	19.3 ± 1.5	8.2	16.5±0.4	72±0
Bonferroni worst-step	11.1 ± 1.7	9.5	23.7±0.7	69±3	7.3 ± 1.1	5.7	16.5±1.1	88±4	12.2 ± 1.4	7.4	18.8±0.3	81±1
coordwise Bonferroni	2.6 ± 0.3	2.6	8.8±0.3	92±0	2.2 ± 0.4	2.1	6.0±0.3	86±1	1.7 ± 0.6	1.3	3.2±0.3	79±2
final-step (UALA-style)	8.7 ± 0.3	8.7	31.6±0.3	100±0	6.2 ± 0.4	6.0	19.6±0.4	100±0	15.8 ± 0.9	9.7	19.8±0.2	66±0
TSC (D=4)†	12.4 ± 1.5	10.2	15.3±0.7	73±3	14.0 ± 1.5	9.0	**12.8±0.5**	**46±1**	20.1 ± 2.2	9.5	**10.6±0.7**	**40±2**
L-TSC (h=0.15)	**14.1 ± 1.7**	9.8	**10.9±1.0**	**55±5**	**14.8 ± 1.8**	10.1	17.1±1.4	55±4	17.6 ± 2.5	8.3	11.2±1.3	43±4

How many labeled reference episodes. TSC reaches its full-reference commit rate with a few dozen labeled reference episodes on most populations, and the two scalars with a reference-fitted component do not degrade at the same rate (Figure 4). The study sub-samples the pre-specified reference partition to $n_{\text{ref}} \in \{25, 50, 100, 150, 200\}$ by five nested draws. Only the reference-dependent components are re-fitted: the two labeled clouds of TSC and L-TSC, the (k, w) grid of the TRACER-tail (ϕ) and the correct-step ECDF of the Bonferroni worst-step typicality. The calibration and test partitions are those of the main grid, so every difference is paired and the full-reference condition reproduces the main-grid cells exactly. The realized CER stays within $\alpha + 1/\sqrt{n_{\text{test}}}$ of the cap at every size at $\alpha = 0.10$.

At $\alpha = 0.10$ TSC is within one point of its full-reference commit rate from $n_{\text{ref}} \leq 50$ on more than half of the populations and from $n_{\text{ref}} \leq 100$ on most. It keeps climbing up to the full reference on MuSiQue and on pooled BFCL with Mistral Small 4. L-TSC needs more references and is not

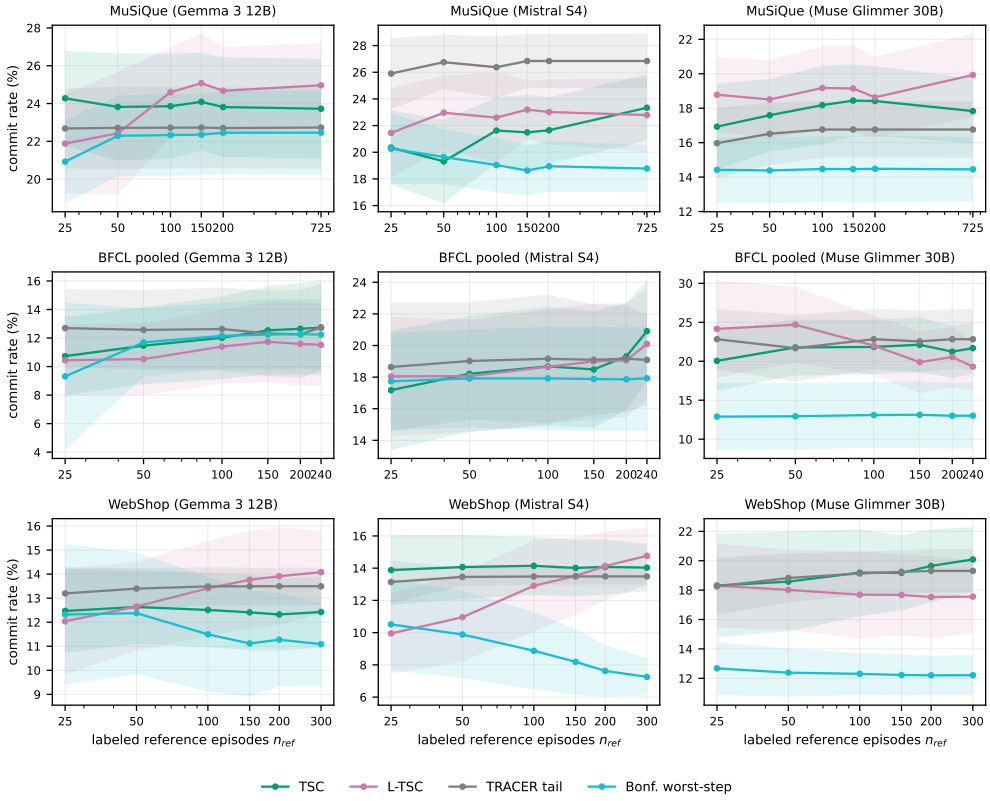

Figure 4: **Commit rate against the number of labeled reference episodes for the four reference-fitted scores ($\alpha = 0.10$, E1).** One panel per population; the horizontal axis is n_{ref} on a log scale, from 25 to the full pre-specified reference partition (725 on MuSiQue, 240 on pooled BFCL, 300 on WebShop). Curves are the mean over the 5×50 (draw, rotation) pairs, bands ± 1 std; the rightmost point of each curve is the main-grid value. The eight scores that fit nothing on the reference are exactly constant in n_{ref} and are not drawn. Two (draw, rotation) pairs at $n_{\mathrm{ref}} = 25$ on WebShop with Gemma 3 and two with Mistral Small 4 held no correct reference and are excluded from the means, never imputed; protocol and readings are in the text. The figure shows the shape of each curve within one population; it ranks no score, and the position of a knee is not comparable across populations, whose labeled clouds differ in size with accuracy.

monotone, because it ranks against length-local neighbours and reads a few references per length. On pooled BFCL with Muse Glimmer it commits more at $n_{\mathrm{ref}} = 25$ than at the full reference. The TRACER-tail (ϕ) saturates early on its coarse grid. The Bonferroni worst-step typicality moves most at $n_{\mathrm{ref}} = 25$ on pooled BFCL with Gemma 3, where it commits less. On the two WebShop populations with long looping episodes it commits *more* with a small ECDF. A narrower step support puts the worst step of more test episodes beyond it, their score saturates at zero and is never committed, and the budget moves from the very long episodes to short ones.

The same four coordinates: aggregation, order, and paths off both clouds. Read on the same four coordinates, TSC commits most on most populations (Table 9; commit rate and placement of the matched pairs in Figure 1). On WebShop it leads every aggregation in commit rate on nearly every development draw. Realized risks are close but not equal (R column), so a small commit gap is no gain at exactly matched risk. Step order is tested by shuffling the steps before interpolation (Appendix D). At $\alpha = 0.10$ this lowers the commit rate of TSC on seven populations, with no consistent sign for L-TSC, so raw order carries a small part of the commit rate of TSC. The spatial rank is not affine invariant (Appendix B). Its sensitivity to the metric is read together with the other factors of the representation in Table 14. Last, a contrast of two spatial ranks is not an outlier detector: a path far from both reference clouds ranks high against both, scores near zero, and falls on

Table 9: **Six scores of the same four coordinates $U = \Phi^\star(\tau)$ on the nine populations ($\alpha = 0.10$, E1).** Per population: commit rate CR (%), with the realized wrong-commit mass R (%) in parentheses. MuSiQue and BFCL: mean $\pm$ std across five reference draws of 50-rotation means; WebShop: over 50 development draws on the fixed test split. Best **bold**, second underlined per column among the six; the L-TSC row below the rule reads $\log T$ and is not ranked. Specification: Appendix C.

MuSiQue			
Score	**Gemma 3 12B**	**Mistral S4**	**Muse Glimmer 30B**
mean of U	24.6 ± 0.6 (10.0)	19.5 ± 0.3 (9.9)	15.1 ± 0.3 (9.8)
max of U	22.5 ± 0.5 (10.0)	21.1 ± 0.4 (9.9)	15.3 ± 0.3 (9.9)
final coordinate U_D	**26.2 ± 0.5 (9.9)**	21.6 ± 0.8 (9.8)	13.5 ± 0.4 (9.8)
RMS of U	24.2 ± 0.5 (10.1)	20.3 ± 0.3 (9.9)	15.0 ± 0.2 (9.9)
pseudo-Noisy-OR on U	24.6 ± 0.6 (10.0)	19.6 ± 0.3 (9.9)	15.0 ± 0.3 (9.8)
TSC	22.6 ± 0.6 (10.0)	**23.1 ± 0.9 (10.0)**	**19.3 ± 1.0 (9.8)**
L-TSC	23.9 ± 0.9 (9.9)	23.1 ± 1.3 (9.9)	20.3 ± 1.4 (9.8)
BFCL pooled			
Score	**Mistral S4**	**Gemma 3 12B**	**Muse Glimmer 30B**
mean of U	18.1 ± 1.1 (9.7)	12.4 ± 0.6 (10.0)	19.6 ± 1.7 (9.8)
max of U	19.6 ± 1.8 (10.0)	12.3 ± 0.8 (10.0)	17.8 ± 1.0 (9.8)
final coordinate U_D	**19.8 ± 0.9 (10.1)**	10.8 ± 0.4 (10.3)	20.1 ± 1.2 (10.0)
RMS of U	18.8 ± 0.9 (9.9)	12.3 ± 0.7 (9.9)	19.4 ± 1.6 (10.0)
pseudo-Noisy-OR on U	18.5 ± 1.1 (9.8)	**12.4 ± 0.6 (10.0)**	19.9 ± 1.8 (9.9)
TSC	19.2 ± 1.5 (10.0)	12.0 ± 0.5 (9.8)	**22.4 ± 1.3 (10.0)**
L-TSC	18.2 ± 1.4 (9.9)	12.1 ± 0.8 (9.9)	24.6 ± 3.1 (9.9)
WebShop (held-out test, 50 development draws)			
Score	**Gemma 3 12B**	**Mistral S4**	**Muse Glimmer 30B**
mean of U	8.2 ± 0.5 (8.0)	7.5 ± 0.5 (7.5)	11.5 ± 1.3 (7.7)
max of U	8.7 ± 0.7 (7.8)	10.2 ± 0.5 (9.3)	13.7 ± 1.0 (8.0)
final coordinate U_D	8.7 ± 0.3 (8.7)	6.2 ± 0.4 (6.0)	15.8 ± 0.9 (9.7)
RMS of U	8.2 ± 0.5 (7.9)	8.5 ± 0.5 (8.5)	11.7 ± 0.7 (8.2)
pseudo-Noisy-OR on U	8.1 ± 0.6 (7.9)	7.5 ± 0.4 (7.5)	11.6 ± 1.6 (7.8)
TSC	**12.4 ± 1.5 (10.2)**	**14.0 ± 1.5 (9.0)**	**20.1 ± 2.2 (9.5)**
L-TSC	14.1 ± 1.7 (9.8)	14.8 ± 1.8 (10.1)	17.6 ± 2.5 (8.3)

whichever side of the calibrated threshold zero lies. On all nine populations that threshold is negative at $\alpha = 0.10$, so such paths abstain; where it is nearest to zero (WebShop with Gemma 3 12B), the few near-zero paths that clear it are mostly wrong.

Where the error mass lands: length groups, category groups, mixture shift. TSC and L-TSC place the permitted errors most evenly (Figure 5 and Tables 10 to 12). The placement is read on a partition that no score observes. The cap constrains only the marginal CER, and no score controls a group. pseudo-Noisy-OR realizes its cap in the shortest group on every population. With Muse Glimmer it commits most of the shortest BFCL group, far above the cap there, and almost nothing in the longest. On WebShop nearly all of its mass lies in the shortest group (H column of Table 8). The mean-family scores point the other way on WebShop. With Gemma 3 the final-step score commits no three-step episode and only long ones, and none of those commits is correct. On MuSiQue with Mistral Small 4 L-TSC keeps its three group rates at the cap. Read against the per-group accuracies of Table 5 (Appendix D), the commit-rate leaders of Table 8 show an observational association: pseudo-Noisy-OR leads where accuracy falls steeply with length, and its margin grows with that gradient.

Over the nine populations TSC and L-TSC have the lowest maximum W, and L-TSC has the lowest maximum H (Table 11). The last column of Table 11 is the risk under the worst length mixture within total-variation radius $\rho = 0.2$ of the source mixture (Corollary A1). Every group holds more than a share ρ of its test episodes, so each of these cells is the attained maximum for the plug-in rates.

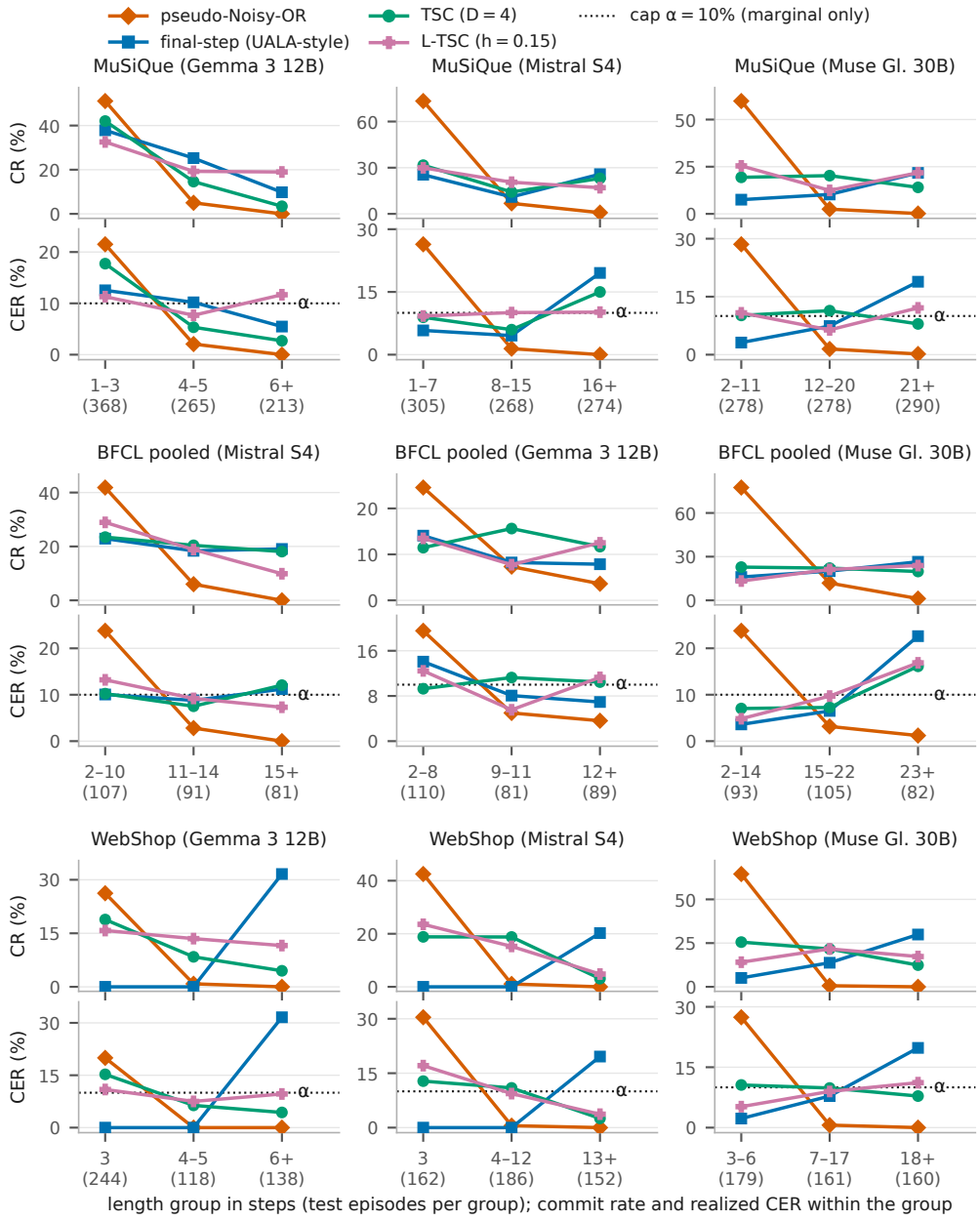

Figure 5: **Where each score commits and errs across the length groups ($\alpha = 0.10$, E1).** One panel pair per population: top, the commit rate inside each length group; bottom, the realized CER inside it, cap dotted (steps; test episodes per group in parentheses). Means over the 50 seeds of each population's protocol (rotations of the pre-specified reference draw on MuSiQue and BFCL, reference/calibration redraws on the fixed WebShop test split); WebShop groups are pre-declared on the development pool. Four scores of the fixed comparator subset of Table 10, which gives the exact values and adds the max and mean step scores and the length-only gate. Each panel has its own vertical scale, so heights compare within a panel, not across populations. The cap constrains only the marginal CER: a within-group CER near the dotted line is a placement reading, no score controls a per-group rate, and a zero CER where a score commits almost nothing (top panel) is abstention, not accuracy.

Under that mixture TSC and L-TSC again have the lowest worst cell over the nine. Max and the TRACER-tail (ϕ) are close behind. The Bonferroni worst-step typicality and pseudo-Noisy-OR trail

Table 10: **Within-group commit rate and CER on the population's length terciles ($\alpha = 0.10$, E1; %).** The six scores of Table 2 and the length-only gate; n = test episodes per group. Means over the 50 seeds of each population's protocol (Appendix D). Per group row, among the six scores: highest CR, and CER closest to the cap among the scores that commit there, in **bold**, second underlined; the gate is not ranked. The cap constrains the marginal CER only.

		mean		max		pNOR		final-step		TSC		L-TSC		gate	
Population	**group (n)**	**CR**	**CER**	**CR**	**CER**	**CR**	**CER**	**CR**	**CER**	**CR**	**CER**	**CR**	**CER**	**CR**	**CER**
MuSiQue (Gemma 3 12B)	1–3 (368)	42.4	16.0	44.1	19.6	**51.1**	21.5	37.9	12.5	42.0	17.7	32.6	**11.3**	23.4	15.1
	4–5 (265)	16.8	6.4	10.3	3.6	5.0	2.1	**25.3**	**10.2**	14.6	5.3	19.3	7.7	0.0	0.0
	6+ (213)	4.5	4.0	1.4	1.4	0.0	0.0	9.8	5.5	3.4	2.7	**19.0**	**11.7**	0.0	0.0
MuSiQue (Mistral S4)	1–7 (305)	12.8	3.6	46.4	14.7	**73.3**	26.4	25.4	5.8	31.6	8.9	29.9	**9.2**	64.4	23.8
	8–15 (268)	7.4	2.0	17.8	8.5	6.8	1.4	11.0	4.6	14.2	6.0	**20.6**	**10.1**	0.0	0.0
	16+ (274)	**35.0**	24.0	14.0	7.2	0.8	0.0	25.8	19.5	23.0	15.0	17.0	**10.2**	0.0	0.0
MuSiQue (Muse Glimmer 30B)	2–11 (278)	7.5	2.7	24.4	11.8	**59.8**	28.5	7.5	3.1	19.4	**10.2**	25.4	10.8	53.5	26.5
	12–20 (278)	8.7	5.1	15.1	**9.8**	2.5	1.5	10.3	7.4	**20.3**	11.4	12.4	6.4	0.0	0.0
	21+ (290)	**26.6**	21.2	11.1	**8.5**	0.2	0.2	21.7	18.9	14.0	8.0	21.8	12.1	0.0	0.0
BFCL pooled (Mistral S4)	2–10 (107)	20.8	11.5	29.2	15.6	**41.9**	23.8	22.9	**10.1**	23.5	10.2	28.9	13.2	37.6	21.5
	11–14 (91)	14.0	6.7	14.7	6.0	6.0	2.8	18.4	8.8	**20.4**	7.5	18.8	**9.2**	0.0	0.0
	15+ (81)	**21.0**	12.0	10.2	6.1	0.0	0.0	19.1	**11.2**	18.1	12.1	9.9	7.3	0.0	0.0
BFCL pooled (Gemma 3 12B)	2–8 (110)	10.2	8.7	16.8	12.4	**24.5**	19.5	14.1	14.1	11.5	**9.3**	13.5	12.4	24.8	21.6
	9–11 (81)	10.4	7.6	11.8	**8.8**	7.3	5.0	8.2	8.1	**15.6**	11.3	7.8	5.5	0.0	0.0
	12+ (89)	**15.9**	14.1	9.5	8.8	3.6	3.6	7.9	6.9	11.7	**10.5**	12.5	11.3	0.0	0.0
BFCL pooled (Muse Glimmer 30B)	2–14 (93)	22.5	4.5	26.8	**10.4**	**77.5**	23.8	16.0	3.6	22.8	7.0	13.3	4.8	80.5	24.5
	15–22 (105)	19.7	7.2	16.4	8.8	11.9	3.2	20.1	6.5	**22.1**	7.3	21.2	**9.7**	0.0	0.0
	23+ (82)	26.2	19.9	11.7	**11.2**	1.2	1.2	**26.5**	22.6	19.8	16.1	23.9	16.9	0.0	0.0
WebShop (Gemma 3 12B)	3 (244)	2.1	1.8	23.2	17.1	**26.2**	20.0	0.0	0.0	18.9	15.3	15.8	**10.9**	0.0	0.0
	4–5 (118)	0.8	0.0	8.3	4.6	0.8	0.0	0.0	0.0	8.4	6.4	**13.5**	**7.5**	0.0	0.0
	6+ (138)	25.6	25.6	0.7	0.7	0.0	0.0	**31.6**	31.6	4.5	4.3	11.6	**9.6**	0.0	0.0
WebShop (Mistral S4)	3 (162)	0.0	0.0	31.0	20.0	**42.5**	30.4	0.0	0.0	18.8	**12.8**	23.6	17.1	0.0	0.0
	4–12 (186)	0.0	0.0	8.7	3.9	1.1	0.5	0.0	0.0	**18.8**	10.9	15.2	**9.4**	0.0	0.0
	13+ (152)	19.6	19.0	0.7	0.7	0.0	0.0	**20.2**	19.6	3.1	2.5	4.8	**3.7**	0.0	0.0
WebShop (Muse Glimmer 30B)	3–6 (179)	7.9	3.9	41.4	16.5	**64.5**	27.4	5.1	2.3	25.6	**10.6**	14.1	5.2	50.8	21.2
	7–17 (161)	4.4	1.9	9.5	4.8	0.6	0.6	13.8	7.8	21.6	**9.8**	**21.7**	9.0	0.0	0.0
	18+ (160)	23.0	18.1	4.4	2.5	0.0	0.0	**29.9**	19.8	12.4	7.8	17.3	**11.2**	0.0	0.0

Table 11: **Nine-population summary ($\alpha = 0.10$, E1).** Left: unweighted mean commit rate over the nine populations (%) and the mean and worst deficit to the per-population leader (points). Right: the score's worst cell over the nine populations: largest within-group CER W, largest single-group share H (%), and largest plug-in risk $R_q = \sum_g q_g r_g$ under a length mixture within total-variation radius $\rho = 0.2$ of the source (Corollary A1). Best **bold**, second underlined per column; the coordinatewise Bonferroni and the length-only gate are not ranked. †pre-specified primary method.

	Commit rate			Worst cell over the nine (↓)		
Score	**mean CR**	**mean def.**	**worst def.**	**max W**	**max H**	R_q (ρ=0.2)
mean step score	15.2	6.2	11.7	25.6	100	14.1
max step score	18.1	3.3	12.1	20.0	87	13.6
pseudo-Noisy-OR	**20.6**	**0.9**	**3.1**	30.4	100	16.1
pseudo-Noisy-OR, len-norm.	15.3	6.2	11.1	25.4	100	14.1
SAUP (ϕ), RMS	15.5	5.9	10.2	24.3	100	14.0
SAUP (ϕ), pos.	15.4	6.0	11.8	31.3	100	15.0
TRACER tail (ϕ)	18.6	2.8	7.7	20.0	87	13.6
Bonferroni worst-step	14.4	7.0	17.5	28.9	88	16.0
coordwise Bonferroni	2.6	18.8	30.5	8.8	92	5.8
final-step (UALA-style)	15.9	5.5	9.9	31.6	100	15.0
TSC (D=4)†	18.5	2.9	8.9	17.7	77	13.1
L-TSC (h=0.15)	18.3	3.1	11.2	**17.1**	**55**	**12.8**
length-only gate	13.4	8.1	16.7	26.5	100	14.0

Table 12: **Error allocation across category groups ($\alpha = 0.10$, E1): worst within-category CER W and largest single-category share H (%), twelve scores on the nine populations.** Same seeds, dispersion and marks as Table 8, over category groups instead of length groups: MuSiQue question type (six), BFCL multi-turn category (four), WebShop product category (five). No category enters any score or threshold. †pre-specified primary method.

MuSiQue	**Gemma 3 12B**		**Mistral S4**		**Muse Glimmer 30B**	
Score	$W\downarrow$	$H\downarrow$	$W\downarrow$	$H\downarrow$	$W\downarrow$	$H\downarrow$
mean step score	15.0±1.2	54±1	12.7±0.6	46±1	20.4±1.0	38±1
max step score	13.5±1.1	57±1	12.9±1.1	55±1	23.5±1.3	29±1
pseudo-Noisy-OR	12.6±0.5	66±1	**12.5±0.4**	65±1	18.0±1.4	54±1
pseudo-Noisy-OR, len-norm.	15.0±1.2	54±1	12.9±0.5	46±1	20.7±1.0	37±1
SAUP (ϕ), RMS	15.9±1.1	53±1	13.1±0.5	45±1	23.4±1.1	35±1
SAUP (ϕ), pos.	16.3±1.4	48±1	15.0±0.6	**36±1**	21.7±1.7	35±1
TRACER tail (ϕ)	13.5±1.1	57±1	12.9±1.1	55±1	23.5±1.3	29±1
Bonferroni worst-step	16.7±1.4	44±1	14.9±0.9	39±1	31.5±2.0	**27±1**
coordwise Bonferroni	10.2±1.0	34±2	7.2±1.3	51±2	3.9±0.7	48±2
final-step (UALA-style)	13.4±0.9	49±1	17.9±1.9	47±1	22.1±2.0	42±1
TSC (D=4)†	**11.7±1.0**	55±1	14.2±0.9	51±1	13.8±0.5	44±1
L-TSC (h=0.15)	15.2±1.1	**43±1**	14.1±0.8	50±1	**12.0±0.9**	51±1
BFCL pooled	**Mistral S4**		**Gemma 3 12B**		**Muse Glimmer 30B**	
Score	$W\downarrow$	$H\downarrow$	$W\downarrow$	$H\downarrow$	$W\downarrow$	$H\downarrow$
mean step score	13.4±1.0	36±2	13.7±1.1	36±2	12.4±1.0	33±2
max step score	13.4±0.9	37±2	14.2±1.1	37±2	11.4±1.2	**28±2**
pseudo-Noisy-OR	14.2±1.3	36±2	15.1±1.0	37±2	13.9±1.0	38±2
pseudo-Noisy-OR, len-norm.	13.5±1.1	36±2	13.8±1.2	37±2	12.7±1.1	34±2
SAUP (ϕ), RMS	14.0±1.2	37±2	14.0±0.9	38±2	11.8±1.0	32±2
SAUP (ϕ), pos.	15.2±1.3	39±2	12.7±1.3	34±2	11.9±0.9	30±2
TRACER tail (ϕ)	13.3±0.9	36±2	14.3±0.8	37±2	11.9±1.0	31±2
Bonferroni worst-step	13.6±1.0	37±2	15.7±0.9	42±2	15.0±1.3	35±2
coordwise Bonferroni	0.0±0.0	—	0.0±0.0	—	0.0±0.0	—
final-step (UALA-style)	12.7±1.2	34±2	15.4±1.0	39±2	11.9±0.7	31±2
TSC (D=4)†	14.7±1.1	40±2	13.7±1.3	36±2	**11.0±0.9**	29±1
L-TSC (h=0.15)	**11.4±0.9**	**30±2**	**12.2±1.2**	**28±2**	13.1±0.9	32±2
WebShop (held-out test)	**Gemma 3 12B**		**Mistral S4**		**Muse Glimmer 30B**	
Score	$W\downarrow$	$H\downarrow$	$W\downarrow$	$H\downarrow$	$W\downarrow$	$H\downarrow$
mean step score	**11.8±0.0**	30±0	9.9±0.3	35±0	9.7±0.0	26±0
max step score	15.5±0.2	33±0	18.2±0.2	41±0	11.7±0.3	30±0
pseudo-Noisy-OR	16.0±0.3	30±0	15.4±0.1	33±1	18.0±0.4	37±0
pseudo-Noisy-OR, len-norm.	**11.8±0.0**	30±0	10.3±0.1	36±0	10.3±0.2	27±0
SAUP (ϕ), RMS	12.2±0.2	29±0	12.3±0.2	39±0	11.8±0.2	29±0
SAUP (ϕ), pos.	14.1±0.5	33±0	9.6±0.1	32±1	9.6±0.4	27±1
TRACER tail (ϕ)	15.5±0.2	33±0	18.2±0.2	41±0	11.7±0.3	30±0
Bonferroni worst-step	15.0±0.4	32±1	**8.1±0.4**	29±1	13.9±0.3	39±0
coordwise Bonferroni	6.4±0.2	49±1	4.7±0.2	40±2	2.3±0.2	37±2
final-step (UALA-style)	12.9±0.1	30±0	8.5±0.2	29±0	20.5±0.2	39±0
TSC (D=4)†	13.0±0.6	**26±1**	11.8±0.7	**26±1**	14.9±0.8	31±1
L-TSC (h=0.15)	13.4±0.8	28±1	15.8±0.8	29±1	**9.5±0.8**	**25±2**

by a few points, although pseudo-Noisy-OR otherwise has the most robust marginal record. Over category groups (MuSiQue question type, BFCL multi-turn category, WebShop product category) TSC or L-TSC has the smallest H and the smallest W on most populations. TSC alone leads W on two.

The shift bound, realized. Table 13 evaluates Corollary A1 seed by seed rather than on seed-averaged groups. The calibrated threshold is kept. The test partition of each seed is re-weighted

Table 13: **Realized wrong-commit mass under the worse of two length shifts, threshold kept ($\alpha = 0.10$, $\rho = 0.20$, E1; %).** For each seed the test partition is re-weighted toward a length-tercile mixture at total-variation distance ρ from the test mixture, ρ moved from the long tercile to the short one or conversely (Corollary A1); the cell is the larger of the two masses, mean over 50 seeds. Superscript: the direction that hurts, S toward short episodes, L toward long ones. Lowest per population in **bold**; last column: maximum over the nine populations.

	MuSiQue			BFCL pooled			WebShop			
Score	Gemma	Mistral	Muse	Gemma	Mistral	Muse	Gemma	Mistral	Muse	**max**
TSC (D=4)	13.1^{S}	11.2^{L}	10.2^{S}	10.5^{L}	10.3^{L}	11.6^{L}	12.3^{S}	11.1^{S}	10.0^{S}	13.1
L-TSC (h=0.15)	$\mathbf{10.3}^{\mathrm{L}}$	$\mathbf{10.0}^{\mathrm{L}}$	$\mathbf{10.1}^{\mathrm{L}}$	$\mathbf{10.3}^{\mathrm{S}}$	11.4^{S}	12.6^{L}	$\mathbf{10.0}^{\mathrm{S}}$	12.8^{S}	$\mathbf{9.5}^{\mathrm{L}}$	12.8
pseudo-Noisy-OR	14.3^{S}	15.2^{S}	15.6^{S}	13.4^{S}	14.8^{S}	13.9^{S}	13.7^{S}	16.1^{S}	15.5^{S}	16.1
max step score	13.6^{S}	11.8^{S}	10.7^{S}	10.9^{S}	11.6^{S}	$\mathbf{10.2}^{\mathrm{L}}$	12.9^{S}	12.0^{S}	11.0^{S}	13.6
final-step score (UALA-style)	11.4^{S}	12.6^{L}	13.1^{L}	11.5^{S}	$\mathbf{10.2}^{\mathrm{L}}$	14.0^{L}	15.0^{L}	$\mathbf{9.9}^{\mathrm{L}}$	13.2^{L}	15.0

by the importance weights q_g/p_g toward a length-tercile mixture q at total-variation distance $\rho \in \{0.05, 0.10, 0.20\}$ from the test mixture p. The quantity read is the realized wrong-commit mass $R_q = \sum_g q_g r_g$. On every (population, score, α, ρ) cell this mass stays below the per-seed bound $R_p + \rho(\max_g r_g - \min_g r_g)$. Under the adversarial mixture it equals the bound, as the identity requires for a fixed rule.

Two directions were fixed a priori, $\rho = 0.20$ moved from the long tercile to the short one and conversely. Each score of the table exceeds α in one of the two directions on nearly every population. The direction that hurts a score is the one that enlarges the tercile carrying its largest share of the realized mass (the S and L marks of Table 13). The two agree on 94 of the 105 (population, score) cells whose largest share sits on an extreme tercile, and on every cell where that share exceeds one half. pseudo-Noisy-OR is the clearest case. It loses under long→short on every population, by about ten points relative to the opposite direction, and its realized mass then exceeds α on nearly every seed. Each scalar loses under one direction on nearly every population. TSC and L-TSC instead split almost evenly, with gaps of a point or less, and their sign follows their largest share. The last column reads the worse of the two directions, maximized over the nine populations. There TSC and L-TSC stay below every scalar score, including the Bonferroni worst-step typicality and the length-only gate outside the table. This is the order of the shift column of Table 11, recovered under two fixed directions.

Sensitivity: configuration and signal. The pre-specified configuration is one member of a stable family. The commit-rate leaders move with the signal, whereas the placement reading does not (Figure 6 and Table 15; signals: Table 6). $D = 4$ and $h = 0.15$ were fixed before the runs, and no configuration is selected from the sweep. The interpolation dimension affects the commit rate of TSC marginally on about half of the populations and by several points on the others. $D = 4$ is not uniformly best. It trails $D = 3$ or $D = 5$ on a few populations. The bandwidth affects the commit rate of L-TSC marginally on most populations and by several points on the rest. The narrowest kernel fails on WebShop with Mistral Small 4, where a large tied block of episodes shares the minimum length and every permitted error then lands in that group. The pre-specified L-TSC lowers H against TSC where the latter concentrates on short paths and raises it elsewhere. Table 15 repeats the full grid under E3, an entropy rather than a chosen-token probability, with every score in its E1 configuration. The placement pattern of Table 8 is preserved: TSC or L-TSC has the lowest W and the smallest H on most populations.

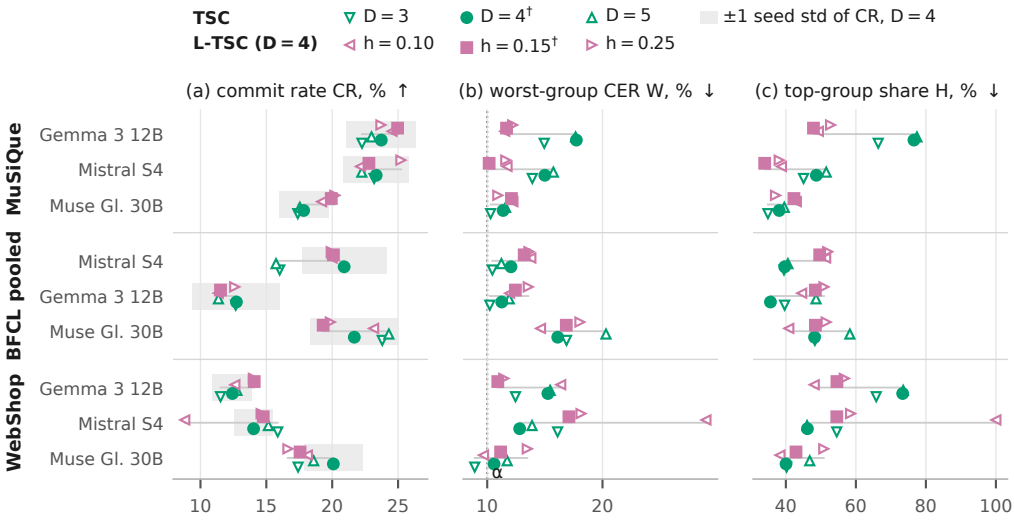

Figure 6: **TSC and L-TSC variant sweep on the nine populations ($\alpha = 0.10$, E1, global CRC).** The six configurations of the sweep: TSC at interpolation dimension $D \in \{3, 4, 5\}$ (green) and L-TSC at D=4 with bandwidth $h \in \{0.10, 0.15, 0.25\}$ on $\log T$ (pink); filled markers and † mark the two configurations fixed before the runs, which are the rows of Table 8. One row per population (generator on the left, benchmark in bold), values are means over the 50 seeds of each population's protocol (rotations on MuSiQue and BFCL, reference/calibration redraws on the fixed WebShop test split); the thin line spans the six configurations. **(a)** commit rate, with the gray band ± 1 seed std of TSC at D=4; **(b)** worst within-group CER W over the population's length groups, cap dotted; **(c)** largest single-group share H of the realized wrong-commit mass. The sweep is a sensitivity analysis and marks no leader: choosing a configuration from it would be test-based selection.

Geometry and interpolation of the representation. No factor of the representation moves the commit rate of TSC or L-TSC, averaged over the nine populations, beyond the seed dispersion of the parent (Table 14; definitions in Appendix D). None yields a gain shared by more than four populations. The per-population intervals are 95% percentile bootstrap intervals over the 50 seeds, paired by seed, with 2,000 resamples. The sweep comprises eleven single-factor variants of the two pre-specified configurations, and everything a variant fits is taken from the reference partition alone. Each variant shifts the mean commit rate by under two points. The construction is therefore not under-tuned. No smoother basis improves on the interpolant sampled on the uniform grid. Cubic B-spline, Legendre and reference-fitted FPCA coefficients each lower the mean commit rate of both parents and raise their mean W and H. Representing the path by its increments instead of its levels lowers the commit rate on five of the nine populations for either parent, with gains on the WebShop populations only, and raises H by several points on average. What the rank reads is the level of the trajectory, not its slope. The value transforms (PIT of the steps or of the coordinates) and the metric (diagonal or full whitening) are neutral in mean. The two whitened metrics move TSC by under two points of commit rate and about three of W on any population. This is the bounded sensitivity of a rank that is not affine invariant (Appendix B). The time warps are neutral in mean but are the least stable factor per population, with shifts of either sign of several points. The sweep was run after the parents were fixed, is not pre-registered, and selects nothing.

Table 14: **Sensitivity of TSC and L-TSC to the geometry and interpolation of their representation, one factor at a time (nine populations, $\alpha = 0.10$, E1).** Δ is variant minus parent in points, paired by seed (50 seeds per population), for TSC at D=4 (left) and L-TSC at h=0.15 (right): ΔCR, mean over the nine populations with its minimum and maximum in brackets; ΔW, ΔH, mean change in worst within-group CER and in largest single-group share. Seed std of the parents' commit rates: 1.5–3.5 points. *increments admitted only under full whitening. Variants: Appendix D; no configuration is selected.

	TSC (D=4)			**L-TSC** (h=0.15)		
Variant (one factor changed)	ΔCR [min, max]	ΔW	ΔH	ΔCR [min, max]	ΔW	ΔH
Projection basis (parent: the interpolant sampled on the uniform D-grid)						
cubic B-spline basis	−0.6 [−3.0, +1.0]	+0.1	+1.7	−0.7 [−3.6, +1.6]	+2.2	+2.9
Legendre basis	−0.5 [−4.1, +1.7]	+0.8	+2.8	−0.4 [−4.5, +2.5]	+1.4	+3.1
FPCA basis (reference-fitted)	−0.9 [−6.4, +0.9]	+1.5	+4.2	−0.1 [−2.2, +2.2]	+0.6	+2.3
Represented function (parent: levels of the interpolant)						
increments	−1.1 [−4.6, +1.5]	+0.4	+3.1	−1.8 [−6.5, +1.8]	+1.8	+4.8
levels + increments, whitened*	−0.5 [−2.8, +1.2]	+1.6	+7.6	−0.4 [−1.9, +1.2]	+1.3	+4.9
Value transform (parent: raw step values)						
PIT of the steps (pooled ECDF)	−0.0 [−1.0, +0.7]	+0.1	+0.5	−0.2 [−1.8, +1.4]	+0.6	+1.0
PIT per coordinate	−0.0 [−1.4, +1.0]	+0.2	+1.3	+0.1 [−0.9, +1.1]	+0.4	+1.1
Time warp of the sampling grid (parent: uniform grid)						
log warp (γ=2)	−0.3 [−2.9, +0.9]	+0.5	+0.5	−0.6 [−2.6, +0.1]	+0.0	−0.1
tail warp (γ=2)	−0.2 [−4.4, +3.8]	+0.9	+3.2	+0.4 [−2.4, +5.0]	+0.9	−0.3
Metric of the rank (parent: Euclidean on the raw coordinates; the metric control of Appendix D)						
diagonal whitening	−0.0 [−0.4, +0.4]	−0.4	−0.2	−0.1 [−0.5, +0.2]	+0.3	+1.8
full whitening (shrinkage 0.05)	−0.1 [−1.0, +1.0]	−0.2	+0.8	−0.2 [−1.2, +0.6]	+0.3	+1.5

Table 15: **Full twelve-score grid on the nine populations under E3 (mean token entropy, Table 6; $\alpha = 0.10$).** Same statistics, seeds, dispersion and marks as Table 8, every score in its E1 configuration. The TRACER tail functional was run under E1 only on MuSiQue (cells —); where it coincides with max, only one of the two rows is ranked. †pre-specified primary method.

MuSiQue	**Gemma 3 12B**				**Mistral S4**				**Muse Glimmer 30B**			
Score	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$
mean step score	24.7 ± 2.1	10.0	16.5±0.6	72±1	18.8 ± 1.7	9.7	23.4±0.7	78±1	14.1 ± 1.7	9.9	21.7±0.7	76±1
max step score	24.4 ± 2.1	10.1	19.8±0.6	86±1	26.0 ± 2.4	10.1	15.2±0.6	54±1	16.8 ± 2.2	9.9	**10.4±0.5**	**35±1**
pseudo-Noisy-OR	24.3 ± 2.0	10.1	21.9±0.7	94±0	**29.1 ± 2.3**	10.0	26.7±0.8	96±0	**20.6 ± 2.1**	9.9	28.5±0.9	95±1
pseudo-Noisy-OR, len-norm.	24.7 ± 2.1	10.0	16.5±0.6	72±1	18.8 ± 1.7	9.7	23.3±0.7	77±1	14.1 ± 1.8	9.8	21.8±0.7	76±1
SAUP (ϕ), RMS	24.1 ± 2.2	10.0	17.2±0.7	75±1	20.2 ± 1.9	10.0	23.1±0.7	75±1	13.9 ± 1.9	9.9	22.3±0.8	77±1
SAUP (ϕ), pos.	25.6 ± 2.3	10.1	15.7±0.7	68±1	17.9 ± 1.7	9.8	24.8±0.7	82±1	12.4 ± 1.9	9.9	23.1±0.9	80±1
TRACER tail (ϕ)	—	—	—	—	—	—	—	—	—	—	—	—
Bonferroni worst-step	23.4 ± 2.3	9.9	12.8±0.6	56±1	19.4 ± 1.9	10.1	22.4±0.7	72±1	13.6 ± 2.1	10.0	22.0±0.8	76±1
coordwise Bonferroni	5.5 ± 1.0	2.3	4.0±0.3	45±2	7.8 ± 1.3	4.0	8.1±0.6	65±2	3.8 ± 1.1	2.6	5.1±0.5	66±3
final-step (UALA-style)	**27.7 ± 2.3**	10.0	12.5±0.5	54±1	21.5 ± 2.0	9.9	18.3±0.6	60±1	12.8 ± 2.1	9.9	18.6±0.8	64±1
TSC (D=4)†	23.7 ± 2.8	10.0	17.9±0.6	78±1	22.0 ± 2.5	10.0	16.5±0.9	53±1	17.9 ± 2.2	9.8	12.0±0.5	40±1
L-TSC (h=0.15)	26.3 ± 2.3	10.3	**11.7±0.6**	**48±1**	24.8 ± 2.4	9.8	**10.4±0.6**	**36±1**	19.0 ± 1.7	9.5	11.6±0.5	42±1

BFCL pooled	**Mistral S4**				**Gemma 3 12B**				**Muse Glimmer 30B**			
Score	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$
mean step score	17.9 ± 3.3	10.3	12.2±0.9	44±2	12.3 ± 2.9	10.0	13.7±1.2	44±2	22.0 ± 4.2	10.0	20.2±1.5	59±2
max step score	18.8 ± 2.7	10.0	15.3±0.8	59±2	12.9 ± 3.4	10.3	11.8±1.2	45±2	19.2 ± 4.2	10.1	**12.4±0.9**	**41±2**
pseudo-Noisy-OR	18.1 ± 3.6	10.1	24.8±1.6	94±1	**13.1 ± 2.9**	10.3	20.2±1.2	77±1	**31.5 ± 4.8**	9.6	24.6±1.6	85±2
pseudo-Noisy-OR, len-norm.	17.8 ± 3.3	10.2	12.3±1.0	44±2	12.3 ± 2.9	10.0	13.6±1.2	43±2	22.0 ± 4.2	10.0	20.2±1.5	59±2
SAUP (ϕ), RMS	17.7 ± 3.5	10.1	11.7±1.0	44±2	12.0 ± 2.8	9.8	12.6±1.2	**41±2**	21.9 ± 3.9	10.3	19.2±1.4	55±2
SAUP (ϕ), pos.	19.3 ± 3.3	10.1	13.0±1.4	38±2	12.6 ± 2.8	10.0	11.9±1.0	43±2	24.1 ± 3.7	10.2	21.8±1.4	63±2
TRACER tail (ϕ)	19.1 ± 2.9	9.8	14.8±0.8	58±2	11.9 ± 2.9	9.8	13.6±1.2	44±2	22.5 ± 3.4	10.2	18.4±1.3	53±2
Bonferroni worst-step	18.7 ± 3.1	10.1	14.6±1.0	42±2	11.8 ± 2.8	10.3	21.6±1.1	66±2	14.1 ± 3.5	10.5	27.4±1.9	77±2
coordwise Bonferroni	0.0 ± 0.0	0.0	0.0±0.0	—	0.0 ± 0.0	0.0	0.0±0.0	—	0.0 ± 0.0	0.0	0.0±0.0	—
final-step (UALA-style)	19.2 ± 3.8	10.2	**10.9±1.2**	39±2	10.4 ± 2.7	10.1	14.2±0.9	55±2	22.0 ± 3.5	10.1	22.1±1.6	64±2
TSC (D=4)†	18.8 ± 3.2	9.5	11.5±1.1	**36±1**	12.1 ± 3.0	9.9	**10.5±0.9**	**41±2**	21.4 ± 3.3	9.6	15.7±1.6	48±3
L-TSC (h=0.15)	**22.3 ± 2.4**	9.8	16.6±0.8	65±2	11.1 ± 2.4	9.8	14.9±0.9	48±2	19.6 ± 3.7	9.9	16.9±0.9	50±2

WebShop (held-out test)	**Gemma 3 12B**				**Mistral S4**				**Muse Glimmer 30B**			
Score	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$	**CR↑**	**CER**	$W\downarrow$	$H\downarrow$
mean step score	9.4 ± 1.1	8.5	26.9±0.2	87±1	5.5 ± 0.5	5.3	17.5±0.4	100±0	12.7 ± 1.3	8.5	19.2±0.4	73±1
max step score	**14.9 ± 0.6**	11.3	18.7±0.1	81±0	11.8 ± 1.2	6.6	16.6±0.6	81±1	20.7 ± 0.7	7.9	14.4±0.1	65±0
pseudo-Noisy-OR	13.1 ± 0.8	9.7	19.9±0.3	100±0	13.1 ± 0.8	8.7	26.8±0.5	100±0	**23.1 ± 1.6**	9.9	27.0±0.7	98±0
pseudo-Noisy-OR, len-norm.	9.4 ± 1.1	8.5	26.8±0.2	87±1	5.5 ± 0.5	5.3	17.5±0.4	100±0	12.7 ± 1.2	8.4	19.0±0.3	72±1
SAUP (ϕ), RMS	11.7 ± 1.1	10.3	25.6±0.2	68±1	6.7 ± 0.6	6.5	21.4±0.5	100±0	12.9 ± 0.9	7.9	15.4±0.3	63±0
SAUP (ϕ), pos.	8.7 ± 0.8	8.7	31.5±0.7	100±0	5.4 ± 1.1	5.4	17.8±1.0	100±0	11.5 ± 1.3	7.5	19.6±0.5	83±0
TRACER tail (ϕ)	14.9 ± 0.6	11.3	18.7±0.1	81±0	11.8 ± 1.2	6.6	16.6±0.6	81±1	20.7 ± 0.7	7.9	14.4±0.1	65±0
Bonferroni worst-step	9.7 ± 2.0	8.7	26.2±0.9	83±4	9.4 ± 1.4	7.9	24.0±1.2	93±3	12.9 ± 1.2	8.8	24.2±0.6	88±1
coordwise Bonferroni	2.8 ± 0.4	2.8	8.8±0.3	86±2	1.9 ± 0.4	1.8	5.9±0.3	99±1	1.7 ± 0.5	1.2	3.6±0.3	92±3
final-step (UALA-style)	8.7 ± 0.5	8.7	31.7±0.4	100±0	6.5 ± 0.5	6.3	20.9±0.5	100±0	16.4 ± 1.3	10.1	20.6±0.4	65±1
TSC (D=4)†	12.6 ± 1.3	9.5	14.7±0.6	75±3	**15.4 ± 1.4**	9.7	**12.8±0.6**	**46±2**	19.3 ± 2.9	8.5	**9.8±0.9**	**41±2**
L-TSC (h=0.15)	13.6 ± 1.6	9.1	**10.6±0.8**	**57±4**	14.2 ± 2.4	9.0	14.9±1.5	53±5	17.0 ± 2.1	7.7	9.9±0.9	**41±3**