LANGUAGE MODELS TRACK STATE BUT UNDERUSE OBSERVATION RELIABILITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Reliable state estimation requires adjusting each update to the quality of new evidence. We study this adjustment in tasks with exact Bayesian references. A five-model survey reveals strong posterior-mean tracking but weak adaptation to sensor reliability. We introduce a paired audit that holds the displayed history fixed and perturbs only the current reading, measuring its influence without regressing on previous model outputs. Under greedy decoding, three checkpoints from two families realize only 5–8% of the prescribed gain change in a matched perturbation test as sensor variance increases sixteenfold. Focused experiments reveal that the same checkpoint can produce accurate updates: supplying the predicted state improves posterior accuracy, and a reasoning-enabled protocol sharply reduces both local-gain and posterior-mean errors on held-out trajectories with unchanged prompts. We translate this diagnosis into the Perception-Dynamics Decoupled Agent (**PDDA**), combining language-based report extraction with explicit Kalman filtering. Complete-system and matched-extraction comparisons improve monitoring accuracy without generated reasoning at the numerical update stage. Reliability-conditioned updating thus provides a diagnostic beyond trajectory agreement and a concrete target for improving language-based estimation.

1 INTRODUCTION

A language model that monitors equipment or combines sensor reports must decide how much each new measurement should change its estimate. The same deviation warrants a larger adjustment when it comes from a precise sensor than from a noisy one. Reasoning, tool use, and memory help agents process reports (Yao et al., 2023; Schick et al., 2023; Packer et al., 2023); reliable estimation also requires weighting those reports against prior uncertainty. This is the classical filtering problem (Kalman, 1960; Anderson & Moore, 1979; Särkkä, 2013), presented through a language interface.

Work on in-context learning shows how transformers can implement posterior inference and learned regression (Xie et al., 2022; Garg et al., 2022; Akyürek et al., 2023; von Oswald et al., 2023). Studies of chat models compare reported beliefs with Bayesian references (Samanta et al., 2026), estimate incoming-signal weights (Wang & Liu, 2026), and fit trajectories with discounted filters (Zhang et al., 2025). Agreement with a posterior trajectory, however, leaves a key ambiguity: an approximately constant observation weight can track well at one noise level. Reliability-sensitive updating requires that this weight also change when sensor quality changes.

Do language models adapt their updates to observation reliability?

We make this distinction measurable with paired reliability interventions. A trajectory-level *effective gain* summarizes the observation weights of five chat models. Our fixed-history audit instead holds the numerical history and center reading constant, perturbs the current reading symmetrically, and measures the change in the reported estimate. Repeating this comparison across declared sensor variances tests whether the same evidence receives the appropriate reliability-dependent weight, with an exact Kalman reference for every context.

This audit identifies a gap that trajectory agreement alone can conceal. Across three checkpoints from the Qwen and Mistral families, a sixteenfold increase in declared sensor variance produces only 5–8% of the gain change required by the reference model. The response moves in the correct

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

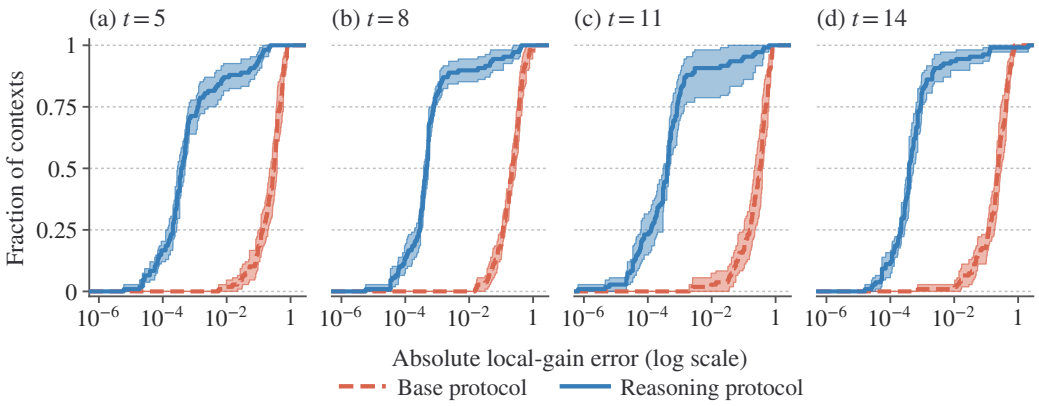


Figure 1: **Local-update accuracy changes with the inference protocol.** Qwen3-8B's cumulative distributions of absolute local-gain error $|\hat{K}_{\text{loc}} - K^*|$ at four update steps. Each panel includes 108 contexts per protocol: 12 held-out trajectories, $R \in \{1, 4, 16\}$, and $\delta \in \{1.5, 3, 6\}$. Curves further left indicate smaller errors; shading shows pointwise 95% trajectory-bootstrap intervals. Prompts and histories match; reasoning mode, temperature, and token budget change together.

direction, but its magnitude is far too small. The paired measurement avoids reusing a noisy previous output on both sides of a regression, directly testing the current update.

Changing stated reliability tests whether a model uses the sensor description; changing observation noise without announcing it tests behavior under model mismatch. Holding readings fixed separates these questions, while counterbalanced schedules prevent noise order from being confounded with history length. These controls isolate the information available at an individual update.

The same checkpoint can also produce accurate updates. In a staged task under reasoning-enabled decoding, supplying the predicted mean and variance sharply improves posterior accuracy. A separate comparison keeps the fixed-history prompts unchanged and evaluates a reasoning protocol on trajectories held out from its development (Fig. 1). Local-gain error falls from 0.301 to 0.019, and posterior-mean error also drops on the same outputs. Budget and representation analyses complement this recovery: successful traces explicitly calculate the update, while changing a noise readout through activation steering does not restore the target gain. Reasoning mode, sampling, and token budget change together; the demonstrated capability belongs to that complete inference protocol.

We connect this diagnosis to monitoring through the Perception-Dynamics Decoupled Agent (**PDDA**). The language model extracts sensor readings from reports; an explicit Kalman filter maintains the numerical belief. This division preserves the model's ability to interpret indirect references and corrections while making reliability-dependent weighting explicit. On telemetry messages, the paired absolute-error reduction is 0.454. Comparisons on identical extractions then separate the numerical update from report reading: a reasoning updater improves accuracy, a code tool matches explicit filtering, and the embedded filter requires no additional language-model generation. Rephrased reports and a public CO_2 sequence test the interface in additional settings.

Our contributions are as follows:

1. **A fixed-history reliability audit.** We combine a five-model trajectory survey with fixed-history perturbations across three checkpoints, using exact context-specific gains to measure the gap between tracking and reliability adaptation.
2. **Recovery within the same checkpoint.** Supplied-state and held-out inference-protocol comparisons demonstrate accurate updating without changing model weights, with improvements in both posterior means and local responses. Budget, probe, and steering analyses complement these tests.
3. **A matched-extraction test of numerical updating.** Complete-system comparisons establish the benefit of extraction–estimation composition; identical extracted streams then separate updater accuracy and generation cost across base, reasoning, code-tool, and embedded filtering alternatives.

2 MEASURING ADAPTATION TO OBSERVATION RELIABILITY

Task and reference posterior. The latent state follows $x_t = x_{t-1} + w_t$, where $w_t \sim \mathcal{N}(0, Q)$, $Q = 1$, and $x_0 \sim \mathcal{N}(0, 2^2)$. Observations satisfy $z_t = x_t + v_t$ with $v_t \sim \mathcal{N}(0, R)$; noises are mutually independent and independent of the past. Each query gives the observation history and asks for a mean and 90% interval in JSON. The Kalman filter supplies the exact posterior $\mathcal{N}(\mu_t^*, P_t^*)$ under the declared linear-Gaussian model. The five-model profile uses Qwen3-1.7B/4B/8B, Mistral-7B-Instruct, and OLMo-2-7B-Instruct. Its base conditions contain 30 to 60 trajectories of T=15 steps under greedy, non-thinking decoding. App. C.2 specifies checkpoint versions and parsing; the focused Qwen3-8B tests have separate prompts and inference settings in App. B.

Fixed-history local response. Our primary behavioral measurement keeps the displayed prefix $h_{<t}$ and the center reading z_t identical across declared reliabilities. For each R and perturbation scale $\delta > 0$, it queries the model at $z_t - \delta$ and $z_t + \delta$. If $f(h_{<t}, z; R)$ is the reported posterior mean, the paired response is

$$K_{\mathrm{loc}}(h_{<t}, z_t, R; \delta) = \frac{f(h_{<t}, z_t + \delta; R) - f(h_{<t}, z_t - \delta; R)}{2\delta}. \tag{1}$$

This finite difference measures the weight assigned to the current reading over a specified range; it is a finite-range response rather than an infinitesimal derivative. We compare its mean at $R \in \{1, 4, 16\}$ with the exact gain $K_t^* = P_t^- / (P_t^- + R)$ for the same contexts. The endpoint gain-adaptation ratio divides the paired change in mean local response by the paired change in exact gain. Confidence intervals resample whole trajectories and preserve the shared histories (App. B.1).

For the exact filter, the posterior mean is affine in the current reading:

$$f_{\mathrm{KF}}(h_{<t}, z; R) = (1 - K_t^*(R))\mu_t^-(h_{<t}, R) + K_t^*(R) z. \tag{2}$$

The predicted state is shared by the two queries, so Eq. 1 returns $K_t^*(R)$ for every $\delta > 0$. Any additive history-dependent offset shared by the two outputs also cancels. The audit thus isolates the reading response from a shared offset; it does not assume that a language model is affine or that all response noise disappears.

Effective gain. We ask how strongly the reported estimate responds to the newest observation. Let $\hat{\mu}_t$ denote the reported mean and define the innovation ν_t as the observation minus the previous estimate. We summarize the response by

$$\hat{\mu}_t - \hat{\mu}_{t-1} = \hat{K}\, \nu_t + \epsilon_t, \qquad \nu_t \coloneqq z_t - \hat{\mu}_{t-1}. \tag{3}$$

A larger $\hat{K}$ gives a larger fitted adjustment per unit of this innovation. Each estimate comes from a fresh query containing the full history; the previous output enters this analysis, not the next base-task prompt. Reporting error in that previous estimate enters both sides of Eq. 3. We therefore treat the coefficient as a descriptive summary and complement it with fixed-context response measurements (App. D.2).

Definition 1 (Effective gain)**.** For trajectories j and steps $t \geq 2$ within one condition, the pooled least-squares coefficient is $\hat{K} = \sum_{j,t} \nu_t^{(j)} \Delta_t^{(j)} / \sum_{j,t} (\nu_t^{(j)})^2$, where $\Delta_t^{(j)} = \hat{\mu}_t^{(j)} - \hat{\mu}_{t-1}^{(j)}$ and the denominator is nonzero. We omit the first update to reduce the influence of the stated prior. Gain confidence intervals resample whole trajectories with the analysis-specific bootstrap settings documented in App. D.

On the recorded exact-filter trajectories, the estimator recovers the steady-state gain within ± 0.02 across the grid. For an unquantized constant-gain recurrence it recovers the coefficient exactly whenever the innovation energy is nonzero (App. D).

Proposition 1 (Reliability determines the optimal gain)**.** *For the scalar system above with $Q > 0$ and $R > 0$, the steady-state Kalman gain is $K^*(R) = \bar{P}/(\bar{P} + R)$, where $\bar{P} = \frac{1}{2}(Q + \sqrt{Q^2 + 4QR})$. It decreases strictly with R, approaches 1 as $R \to 0$, and approaches 0 as $R \to \infty$.*

For the recurrence $\hat{\mu}_t = (1 - K)\hat{\mu}_{t-1} + K z_t$ with constant gain $K \in (0, 1]$ and finite initial MSE, the steady-state MSE relative to x_t, $S(K) = [(1 - K)^2 Q + K^2 R]/[K(2 - K)]$, is uniquely minimized at $K = K^*(R)$. The proof is in App. L.

Across $R \in [0.25, 32]$, the optimal gain decreases from 0.83 to 0.16. Keeping $K = 0.45$ instead incurs steady-state MSE approximately 2.4 and 1.9 times the minimum at the two endpoints.

Connection to discounted filtering. A static discounted filter inflates variance by $1/\gamma$, with $0 < \gamma < 1$, and omits process noise: $P_t^- = P_{t-1}/\gamma$, $P_t = (\gamma/P_{t-1} + 1/R)^{-1}$. Its positive fixed point is $P^* = (1-\gamma)R$, giving $K_{\text{disc}} = (P^*/\gamma)/(P^*/\gamma + R) = 1 - \gamma$, independent of R (Zhang et al., 2025). This explains how accurate trajectory fits can arise from a rule whose steady-state gain does not depend on R. It is a structural comparison, not an identified attention mechanism: our experiments have $Q = 1$, and the paired audit tests against that model's gain (App. B.6).

Reliability interventions. The *stated-noise* intervention changes the sensor noise described in the prompt; the *experienced-noise* intervention changes the observations without announcing a different sensor model. The dense sweep and cross-trajectory factorial share $x_{1:T}$ and $\varepsilon_{1:T}$, using $z_t = x_t + \sqrt{R_t}\,\varepsilon_t$. The within-trajectory factorial shares latent paths and balances block order, with observations paired within each actual-noise condition. Its declared schedules include per-line standard deviations. The fixed-history intervention holds the numerical readings themselves constant while changing declared R. Apps. C.4 and F specify the contrasts.

Definition 2 (Gain-adaptation ratio)**.** For low- and high-noise regimes with $K^*_{\text{lo}} > K^*_{\text{hi}}$, define $\text{GAR} = (\hat{K}_{\text{lo}} - \hat{K}_{\text{hi}})/(K^*_{\text{lo}} - K^*_{\text{hi}})$. A ratio of 1 denotes the full prescribed change, 0 denotes no change, and a negative value denotes a change in the opposite direction. Constant-noise comparisons use steady-state reference gains. Within-trajectory comparisons use the mean Kalman gain of each phase on the same schedule (App. D.3).

Protocol status. The original behavioral, rephrasing, and system tests use frozen protocols and decision criteria. Appendix C.5 distinguishes these from the later fixed-history, staged-state, and inference-protocol analyses.

3 Finding 1 State tracking does not imply gain adaptation

Reported means track the state. At R=4, correlations with the exact posterior mean range from 0.927 to 0.977 across the five models (Table 1A). For the Qwen models and Mistral, Eq. 3 gives reported trajectory-fit R^2 values from 0.72 to 0.83; the corresponding range for OLMo is 0.42 to 0.51. These fits summarize successive reports and motivate testing whether the effective gain changes with noise.

Gain changes are small relative to the reference. In Qwen3-8B's paired eight-point renderer-A sweep, the optimal gain falls from 0.83 to 0.16, while the fitted gain remains between 0.42 and 0.48. Its span of 0.064 is about 10% of the required change, and the confidence intervals exclude the optimal gain at all eight points (Table 11). These fitted coefficients lie below the steady-state reference at low noise and above it at high noise. The fixed-history test below measures the response to the current reading directly. Trajectory error and the descriptive constant-gain comparison appear in App. E.

Across the five-model profile, the reference gain decreases by 0.40 from $R = 1$ to 16, while every observed absolute change is at most 0.102. Qwen3-1.7B even moves in the opposite direction, from 0.51 to 0.58. Gain levels also differ across models: OLMo remains at or below the reference at all tested levels (Table 1A). The common finding is insufficient reliability adaptation, rather than a universal coefficient near 0.45.

The response gap persists across models at fixed histories. The Qwen3-8B audit uses 30 trajectories, four time points, three declared variances, and three perturbation scales; all $2{,}160$ responses parse. At $\delta = 3$, local gains at $R = 1, 4, 16$ are 0.640, 0.638, and 0.619, versus exact gains 0.618, 0.391, and 0.221. The endpoint span is 0.021 $[0.003, 0.040]$, giving GAR 0.054 $[0.008, 0.101]$. Across perturbation scales, mean GAR remains below 0.12 (App. B.1).

The same $\delta = 3$ audit on Qwen3-4B and Mistral-7B gives spans 0.023 and 0.031, respectively (Table 1B). Across these three checkpoints, local responses realize only 5–8% of the exact 0.397 gain change. The positive spans show a correctly directed but severely insufficient response, extending the diagnosis across model scales and families. A matched two-decimal reading check gives Qwen3-8B gain 0.635, close to 0.638 with eight decimals (App. B.4).

Table 1: **Trajectory agreement and multi-model fixed-history responses.** (A) Profiles match stated and actual noise (r at $R = 4$). Outage is the three-step interval-width ratio; order is terminal-mean spread in reference posterior standard deviations. (B) Each model uses $\delta = 3$ and 120 paired contexts per variance. Brackets are trajectory-bootstrap 95% intervals.

A. Archived trajectory profile

		Pooled gain $\hat{K}_R$						
Model	r	$R = 1$	$R = 4$	$R = 16$	$\hat{K}_1 - \hat{K}_{16}$	R^2 range	Outage	Order
Qwen3-1.7B	0.962	0.51	0.55	0.58	−0.07	0.72–0.76	1.46	1.35
Qwen3-4B	0.975	0.50	0.41	0.47	+0.03	0.72–0.78	1.04	0.88
Qwen3-8B	0.977	0.49	0.43	0.45	+0.04	0.73–0.77	1.07	0.68
Mistral-7B	0.972	0.65	0.56	0.55	+0.10	0.79–0.83	1.13	1.29
OLMo-2-7B	0.927	0.25	0.26	0.22	+0.03	0.42–0.51	1.09	0.65
Kalman reference	1.000	0.62	0.39	0.22	+0.40	1.00	1.71	0.00

B. Paired local response at fixed displayed histories

Model / reference	$R = 1$	$R = 4$	$R = 16$	Signed span [95% CI]	Signed GAR [95% CI]
Qwen3-8B	0.640	0.638	0.619	0.021 [0.003, 0.040]	0.054 [0.008, 0.101]
Qwen3-4B	0.584	0.572	0.561	0.023 [0.004, 0.043]	0.058 [0.010, 0.108]
Mistral-7B	0.682	0.670	0.651	0.031 [0.008, 0.055]	0.078 [0.020, 0.139]
Exact Kalman	0.618	0.391	0.221	0.397	1.000

Uncertainty responds too little to missing observations. In the exact filter, posterior variance grows during a gap because the latent state continues to move while no new measurements constrain it. For Qwen3-8B at R=4, the reported 90% interval is already narrower than the reference, with widths 3.12 and 4.19. During a three-step gap, it widens by 1.07× instead of 1.71×, and only 57% of trajectories show any widening (Table 1A). Across all five models, widening factors range from 1.04 to 1.46. A separate static-state control sets Q=0, making the exact posterior invariant to observation order. Permuting the same observations nevertheless changes terminal model estimates by 0.65 to 1.35 posterior standard deviations. Empirical state coverage is 72.5% at $R = 1$ and 61.4% at $R = 16$ for Qwen3-8B's nominal 90% intervals (Table 18).

A fixed discount does not explain the full reported belief. The discounted recurrence above produces a reliability-invariant steady-state gain, but predicts interval widths proportional to $\sqrt{R}$. Reported Qwen3-8B mean widths at $R = 1, 4, 16$ are 2.22, 3.12, and 3.98: a 1.79× endpoint increase, compared with 4× under fixed discounting. For the task's $Q = 1$ Kalman reference, the exact steady-state width ratio is 2.39× (App. B.6). Thus a constant-gain account of the means does not also reproduce the measured uncertainty response; both quantities are needed to assess a reported belief.

Complementary checks test other aspects of the update. A disjoint-vocabulary rendering raises Qwen3-8B's gain level but gives a signed endpoint span of −0.050 (App. E). In a five-room HMM, the model realizes 0.435 of the exact posterior's average belief movement with the more accurate sensor and 0.556 with the less accurate sensor (Table 20). These ratios compare movement with each condition's own reference, not absolute movement between conditions.

Finding 1. Posterior-mean tracking and reliability-conditioned updating are distinct. Across five models the trajectory-level gain changes little with sensor variance; across three checkpoints from two families, the fixed-history response realizes only 5–8% of the exact gain change.

4 Finding 2 Declared reliability is a distinct intervention

Reliability can be conveyed by a sensor description or by variation in the readings. We distinguish the two using paired interventions.

Stated and experienced noise answer different questions. Crossing $R_{\text{prompt}} \in \{1, 16\}$ with $R_{\text{data}} \in \{1, 16\}$ shares latent trajectories and standardized noise across four cells (Table 13). Changing only stated variance gives Qwen3-8B pooled GAR 0.007 $[-0.02, 0.03]$; Mistral gives 0.025. Changing data noise without announcing a new sensor instead tests an unannounced mismatch. A separate fixed-display-history sweep varies stated variance over $128\times$ and gives gain spans 0.033 for Qwen3-8B and 0.033 for Qwen3-30B-A3B (App. F). These archived slopes and the paired reading differences in Finding 1 provide complementary tests.

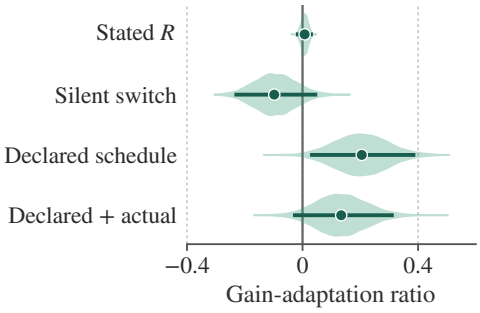


Figure 2: **Responses to reliability inputs.** Bootstrap GAR distributions; dots and bars show pooled estimates and 95% intervals.

Figure 2 summarizes these contrasts with Qwen3-8B's trajectory-bootstrap sampling distributions, drawn with equal maximum widths. Stated R uses the factorial contrast; the other conditions use counterbalanced schedules. GAR normalizes each pooled gain contrast by its reference contrast.

A declared schedule elicits a measurable but partial response. The counterbalanced experiment uses 30 trajectories of 20 steps with low- and high-noise blocks in balanced order. A silent data switch gives $\text{GAR} = -0.098$ $[-0.23, 0.04]$. Declaring the schedule and marking each reading while keeping actual noise constant gives 0.205 $[0.03, 0.38]$; changing both declared and actual noise gives 0.134 $[-0.03, 0.31]$. The declared-schedule contrast shows that the model can respond to explicit reliability schedules, with an update change smaller than the exact reference. An exploratory low-noise-first condition gave GAR 0.654 $[0.51, 0.79]$, with noise level changing together with history length. The counterbalanced schedules separate these factors and yield the smaller responses in Fig. 2; Table 15 retains both protocols.

> **Finding 2.** Paired interventions distinguish a declared sensor model from unannounced changes in data noise. Under the base protocol, a line-by-line declared schedule can elicit some adaptation, while the fixed-history test isolates the small response to stated reliability at one update.

We next vary the information and inference procedure available at the update. This tests whether the weak base response is also the strongest response the same checkpoint can produce.

5 Finding 3 The same checkpoint can produce accurate updates

The base audit tests Qwen3-8B under greedy, non-thinking decoding. We now vary the numerical state supplied to the model and the inference procedure used to process an unchanged prompt.

Supplying the predictive state improves posterior accuracy. The staged task evaluates 240 contexts under reasoning-enabled decoding. A supplies the observation history; B additionally supplies the exact predicted mean μ_t^- and variance P_t^-; C also supplies K_t^*. All 720 outputs parse. Normalized posterior-mean error falls from 0.0226 in A to 0.000077 in B and 0.0000097 in C; the paired A-to-B reduction is 0.0225 $[0.0084, 0.0412]$ (Table 4). Predictive-state information therefore resolves much of the error in this task. Under the separate non-thinking prompt comparisons, formulas and worked examples leave gains around 0.47–0.54, above the $R = 16$ reference 0.221 (Table 16).

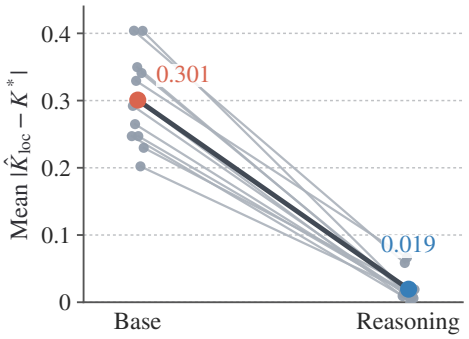


Figure 3: **Held-out local-gain error.** Thin lines pair 12 trajectories (36 contexts each); the heavy line shows their means.

Unchanged prompts support accurate updates under reasoning. We reuse the paired prompts and readings on 12 trajectories held out from protocol development. All 864 calls parse, yielding 432 local-response pairs. Gain MAE falls from 0.301 to 0.019, improving on every trajectory (Fig. 3); the paired reduction is 0.281 $[0.248, 0.315]$. On these same outputs, posterior-mean MAE falls from 1.325 to 0.056, with a paired reduction of 1.270 $[1.069, 1.480]$ and improvements at every tested R (Table 5). The recovery therefore extends beyond the response slope to the posterior level. Reasoning mode, temperature, and budget change together (App. B.3).

A development-set budget sweep holds reasoning temperature at 0.6. Reported local-gain MAE falls from 0.184 at 1,024 tokens to 0.072 at 2,048, 0.022 at 4,096, and 0.019 at 8,192. At 512 tokens, 64% of calls fail to parse; error estimates must therefore be read alongside validity. Audited successful traces explicitly compute predicted variance, gain, and the innovation update (App. B.5).

Decodable noise information does not guarantee an accurate update. A residual-stream probe achieves $R^2 = 0.978$ within seen noise levels but 0.42 on held-out levels; the leading principal component (63% of variance) distinguishes numerical formats. Steering along within-format noise-centroid directions at layer 28 shifts frozen probe predictions by up to $25 \log_2 R$ units while changing gain by less than 0.019, without restoring the calibrated response (App. I). The large readout shift is an extrapolated probe value, not a calibrated change in the model's belief. These interventions distinguish accessible noise features from the computation needed to use them.

Finding 3. Qwen3-8B's update becomes accurate when the predicted state is supplied. With unchanged prompts, a held-out reasoning protocol improves both local responses and posterior means. Probe decodability and the tested activation shifts alone do not recover that response.

The behavioral diagnosis identifies a numerical operation worth making explicit in a monitoring system. We now test whether language-based reading extraction can be paired with a Kalman filter that maintains the state and weights each observation by its declared reliability.

6 Separating evidence extraction from state estimation

PDDA pairs language-based observation extraction with explicit Kalman filtering. Complete systems receive the same message streams; matched-extraction comparisons then isolate the numerical updater.

Monitoring task. Two intermittent sensors with known variances $R_A = 1$ and $R_B = 9$ report a drifting state over 12 steps. Messages include abbreviations, indirect sensor references, corrections, and irrelevant text. We evaluate posterior means at $t \in \{4, 8, 12\}$ on 60 trajectories and terminal alarms $P(x_{12} > 4) > \frac{1}{2}$ (App. K).

6.1 Extraction, estimation, and decision

Each message yields a sensor-value pair $\{\texttt{sensor}\ s,\ \texttt{value}\ v\}$ or no observation (Table 23). Given $(Q, \{R_s\})$, the Kalman recurrence maintains (μ_t, P_t): sensor variance determines each measurement's weight, and uncertainty grows during missing reports. Independent readings at the same time can be incorporated in either order. A Gaussian terminal alarm at probability threshold one half reduces to $\mu_{12} > 4$. We score extraction, posterior-mean error, and interval agreement separately; the telemetry interval target is the reference posterior mean (App. H).

6.2 How extraction errors differ from gain mismatch

An explicit recurrence gives an isolated reading error a predictable influence. At fixed $K = 0.45$, error δ first shifts the estimate by 0.45δ; each subsequent update retains 0.55 of that shift. A mismatched gain instead changes every update. App. L proves both statements, with all other readings and gains held fixed. The complete-system evaluation measures omissions, sensor misidentification, and repeated extraction errors rather than assuming that extraction failures are isolated or independent.

Table 2: **Separating report reading from state estimation.** LLM denotes Qwen3-8B. Telemetry errors use the oracle posterior mean. CO_2 Dev. uses the oracle-parse mean; CO_2 MAE uses the observed series (descriptive). All update rules in (B) share LLM extractions. Bold marks best non-oracle telemetry and deviation values within each panel, including ties.

	Assignment		Telemetry			CO_2 anchor	
System	Reader	Updater	MAE	Median	Alarm	Dev.	MAE
A. Full set: 60 *streams (*180 *checkpoints),* 76 *CO_2 windows*							
End-to-end	LLM	LLM	0.471	0.301	0.98	0.398	0.723
Regex → Kalman	Regex	Kalman	0.211	0.023	**1.00**	0.170	0.689
PDDA	LLM	Kalman	**0.017**	**0.000**	**1.00**	**0.032**	0.605
Oracle parse	Oracle	Kalman	0.000	0.000	1.00	0.000	0.609
B. Held-out half: 30 *streams (*90 *checkpoints),* 38 *CO_2 windows*							
Base LLM	LLM	Single-pass	0.435	0.278	0.98	0.362	0.708
Reasoning LLM	LLM	Reasoning	0.038	0.012	**1.00**	0.045	0.621
Code-tool (PoT)	LLM	Python Kalman	**0.028**	**0.000**	**1.00**	**0.038**	0.617
PDDA	LLM	Kalman	**0.028**	**0.000**	**1.00**	**0.038**	0.617
Single gain	LLM	Constant K	0.736	0.556	n/a	0.960	1.128
Two sensor gains	LLM	K_A, K_B	0.120	0.070	n/a	0.592	0.837
Static discount	LLM	Fixed γ	0.091	0.032	n/a	0.090	0.597
Oracle parse	Oracle	Kalman	0.000	0.000	n/a	0.000	0.627

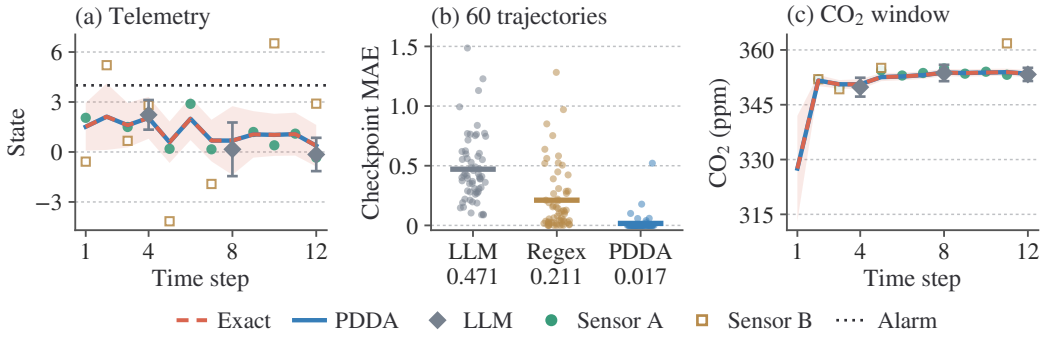


Figure 4: **Explicit estimation improves tracking from language reports.**

6.3 Evaluating extraction and tracking

Report extraction is accurate but not perfect. Qwen3-8B's recall against stored readings is 0.989, versus 0.866 for the audited regex and 0.999 for a template-aware parser. Its unmatched-reference and extra-extraction rates are 1.1% and 0.3%. App. K gives the numerical matching rule and the message-integrity audit.

Explicit estimation reduces complete-system error. End-to-end checkpoint MAE is 0.471, versus 0.017 for PDDA, with a paired absolute reduction of 0.454 [0.386, 0.528] over the same 60 streams (Table 2A). Systems use their own call schedules. Terminal alarm agreement is 1.00 for PDDA and 0.98 end to end; the main improvement is in the numerical belief. Figure 4 shows median-error telemetry and CO_2 examples and the distribution of trajectory errors. Bands are exact 90% posterior intervals; diamonds and error bars are LLM estimates and reported intervals. Each central point represents one trajectory; bars show means.

Identical extractions expose the updater's contribution. On the same 30 held-out extracted streams, posterior-mean MAE is 0.435 with a base LLM, 0.038 with reasoning, and 0.028 with either code-tool or embedded Kalman updating (Table 2B). Reasoning uses about 7,850 output tokens per update; the code tool uses about 180; the embedded filter requires none. Reported updater times are about 14.2 s, 0.45 s, and less than 1 ms, respectively, excluding shared extraction (App. B.7).

Fitted rules use the same extracted streams and a disjoint fitting half. A single gain yields 0.736 MAE, two sensor-specific gains 0.120, and static discounting 0.091. The dynamic discounted family selects $\gamma^* = 1.00$, recovering Kalman filtering and its 0.028 error. This comparison tests whether an inexpensive fitted recurrence suffices after extraction; the full-set 0.017 and held-out 0.028 scores have different denominators.

Additional settings test the interface and sensor assumptions. Rephrasing stored reports with disjoint vocabulary gives a PDDA-to-end-to-end error ratio 0.0002, below the registered 0.25 threshold. On the weekly Mauna Loa CO_2 series distributed with statsmodels (Seabold & Perktold, 2010), PDDA reaches full-set series MAE 0.605, close to 0.609 with oracle readings. On held-out windows, however, static discounting gives 0.597 versus Kalman's 0.617: optimality under the synthetic Gaussian model does not guarantee the best fit to an observed series. Finally, assuming $R_B/3$ or $3R_B$ gives telemetry MAE 0.265 and 0.114, both below end-to-end 0.471 (App. K).

Finding 4. Separating report reading from numerical estimation reduces telemetry MAE by 0.454 $[0.386, 0.528]$. With extraction fixed, embedded Kalman filtering matches the code tool's accuracy and avoids additional language-model generation.

7 Related work

Bayesian behavior in language models. In-context learning can implement posterior inference and regression (Xie et al., 2022; Garg et al., 2022; Akyürek et al., 2023; von Oswald et al., 2023); related work studies dynamical data (Li et al., 2023) and Kalman-like estimation (Akram & Vikalo, 2026). Chat-model posterior comparisons (Samanta et al., 2026), signal-weight fits (Wang & Liu, 2026), and discounted filters (Zhang et al., 2025) characterize trajectory behavior. Our audit holds displayed histories fixed and compares paired reading responses across sensor reliabilities, separating reliability sensitivity from trajectory fit and revealing protocol-dependent recovery. It complements martingale and asymmetric-update tests (He et al., 2025; Schubert et al., 2024).

Uncertainty and representations. Studies of verbalized confidence (Kadavath et al., 2022; Lin et al., 2022; Tian et al., 2023; Xiong et al., 2024) and conservative belief revision (Edwards, 1968; Tversky & Kahneman, 1974) motivate checking numerical uncertainty. Probes can recover posterior information (Sarfati et al., 2026), and activation additions can alter behavior (Rimsky et al., 2024). Our interventions test whether accessible noise features control the update.

Memory, tools, and estimation. Agents retain reports through reasoning histories and memory (Yao et al., 2023; Shinn et al., 2023; Packer et al., 2023; Park et al., 2023; Hao et al., 2023); tools delegate computation (Schick et al., 2023; Gao et al., 2023; Chen et al., 2023). Language-model forecasting (Gruver et al., 2023; Jin et al., 2024), time-series foundation models (Das et al., 2024), and learned state estimation (Liu et al., 2025) provide other interfaces to sequences. PDDA applies classical measurement processing and state estimation (Kalman, 1960; Anderson & Moore, 1979), evaluating extraction and numerical updating separately.

Scope and negative results. The tasks provide exact references for scalar Gaussian dynamics and a small HMM; they do not establish general-purpose Bayesian reasoning. Mistral fails the registered rephrased-report transfer criterion, held-out-level probing misses its criterion, and a separate $1{,}024$-token trajectory-reasoning run rarely parses. These outcomes, including the distinct reasoning protocols, are retained in Apps. C.5 and M.

8 Conclusion

Language models can track posterior means while underusing the reliability of new evidence. Fixed-history perturbations establish this gap across three checkpoints without regressing on previous model outputs. Supplied predictive state and a held-out reasoning protocol demonstrate that the same checkpoint can execute more accurate updates; probe readouts alone do not ensure that behavior. In monitoring, combining language extraction with explicit filtering improves posterior accuracy and avoids generated reasoning at each numerical update. Reliability-conditioned updating is therefore a measurable capability beyond trajectory agreement, and its diagnosis supports concrete improvements in language-based state estimation.

AI USE STATEMENT

AI-assisted tools were used primarily for manuscript organization, writing, and grammatical polishing, and assisted with figure preparation and consistency checks. The authors manually reviewed the text, mathematical arguments, references, and reported results and take responsibility for the final manuscript.

REPRODUCIBILITY STATEMENT

Section 2 defines the task and evaluation measures; Appendix L gives the mathematical derivations. Appendix C records prompts, model checkpoints, decoding settings, and paired experimental designs. Appendix B details the fixed-history, staged-state, and held-out inference-protocol tests; Appendices I and K describe the internal-state and monitoring experiments. The accompanying source package contains the manuscript, frozen vector figures, tabulated results, and the paired holdout reanalysis script with per-call, per-pair, and per-trajectory data. Its README specifies the build and reanalysis commands and distinguishes this available material from the full model-inference pipeline.

REFERENCES

Usman Akram and Haris Vikalo. Transformers as implicit state estimators: In-context learning in dynamical systems. *Transactions on Machine Learning Research*, 2026.

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? Investigations with linear models. In *International Conference on Learning Representations*, 2023.

Brian D. O. Anderson and John B. Moore. *Optimal Filtering*. Prentice-Hall, 1979.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Transactions on Machine Learning Research*, 2023.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *International Conference on Machine Learning*, 2024.

Ward Edwards. Conservatism in human information processing. In Benjamin Kleinmuntz (ed.), *Formal Representation of Human Judgment*, pp. 17–52. Wiley, 1968.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. PAL: Program-aided language models. In *International Conference on Machine Learning*, 2023.

Shivam Garg, Dimitris Tsipras, Percy Liang, and Gregory Valiant. What can transformers learn in-context? A case study of simple function classes. In *Advances in Neural Information Processing Systems*, 2022.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew Gordon Wilson. Large language models are zero-shot time series forecasters. In *Advances in Neural Information Processing Systems*, 2023.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. Reasoning with language model is planning with world model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2023.

Zhonghao He, Tianyi Qiu, Hirokazu Shirado, and Maarten Sap. Martingale score: An unsupervised metric for Bayesian rationality in LLM reasoning. In *Advances in Neural Information Processing Systems*, 2025.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-LLM: Time series forecasting by reprogramming large language models. In *International Conference on Learning Representations*, 2024.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Rudolf E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

C. D. Keeling and T. P. Whorf. Atmospheric CO2 concentrations derived from flask air samples at sites in the SIO network. In *Trends: A Compendium of Data on Global Change*. Carbon Dioxide Information Analysis Center, Oak Ridge National Laboratory, U.S. Department of Energy, 2004.

Yingcong Li, Muhammed Emrullah Ildiz, Dimitris Papailiopoulos, and Samet Oymak. Transformers as algorithms: Generalization and stability in in-context learning. In *International Conference on Machine Learning*, 2023.

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*, 2022.

Shiqi Liu, Wenhan Cao, Chang Liu, Zeyu He, Tianyi Zhang, Yinuo Wang, and Shengbo Eben Li. One filters all: A generalist filter for state estimation. In *Advances in Neural Information Processing Systems*, 2025.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards LLMs as operating systems. *arXiv preprint arXiv:2310.08560*, 2023.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 2023.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 2024.

Ankur Samanta, Akshayaa Magesh, Tal Lancewicki, Ayush Jain, Youliang Yu, Paul Sajda, Kaveh Hassani, Aditya Modi, Daniel R. Jiang, and Yonathan Efroni. BayesBench: Evaluating LLM belief trajectories under multi-turn evidence accumulation. *arXiv preprint arXiv:2606.30850*, 2026.

Raphaël Sarfati, Eric Bigelow, Daniel Wurgaft, Siddharth Boppana, Jack Merullo, Atticus Geiger, Owen Lewis, Tom McGrath, and Ekdeep Singh Lubana. The shape of beliefs: Geometry, dynamics, and interventions along representation manifolds of language models' posteriors. *arXiv preprint arXiv:2602.02315*, 2026.

Simo Särkkä. *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*, 2023.

Johannes A. Schubert, Akshay Kumar Jagadish, Marcel Binz, and Eric Schulz. In-context learning agents are asymmetric belief updaters. In *International Conference on Machine Learning*, 2024.

Skipper Seabold and Josef Perktold. statsmodels: Econometric and statistical modeling with Python. In *Proceedings of the 9th Python in Science Conference*, 2010.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, 2023.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, et al. 2 OLMo 2 Furious. *arXiv preprint arXiv:2501.00656*, 2025.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2023.

Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157):1124–1131, 1974.

Johannes von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning*, 2023.

Yu Wang and Xiangchen Liu. Bridging expectation signals: LLM-based experiments and a behavioral Kalman filter framework. *arXiv preprint arXiv:2601.17527*, 2026.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit Bayesian inference. In *International Conference on Learning Representations*, 2022.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can LLMs express their uncertainty? An empirical evaluation of confidence elicitation in LLMs. In *International Conference on Learning Representations*, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

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*, 2023.

Jensen Zhang, Jing Yang, and Keze Wang. Large language models as discounted Bayesian filters. *arXiv preprint arXiv:2512.18489*, 2025.

A EVIDENCE AND INTERPRETATION

Reliability as an evaluation target. Paired reliability interventions test whether observation weights change as the reference model predicts. Pooled effective gain summarizes successive full-history reports, while fixed-context perturbations measure the response to the same current reading on the same displayed histories. At $\delta = 3$, the new Qwen3-8B local response changes from 0.640 to 0.619 between declared variances 1 and 16, against a reference change from 0.618 to 0.221. App. D.2 reports a separate archived five-point protocol with different prompts and histories.

Representations and behavior. Probe and steering experiments compare accessible noise information with the resulting update. The within-level and held-out-level splits, numerical-format analysis, and random-direction controls characterize the readout. Steering changes that readout without restoring the target gain in the tested conditions (App. I). The intervention uses within-format noise-centroid chords; large extrapolated probe shifts are not calibrated semantic changes in sensor reliability.

Task and inference conditions. The linear-Gaussian and discrete-state tasks provide exact references under known observation models. The five-model comparison uses greedy, non-thinking inference, with model and decoding settings specified in App. C.2. The staged and held-out protocol comparisons use the Qwen3-8B settings in App. B. The public CO_2 test extends the system comparison to an observed state sequence with artificial sensor noise and generated reports.

From diagnosis to application. PDDA assigns the reliability-dependent update to a classical Kalman filter and uses the language model to extract observations. Comparing update rules on identical extractions isolates the contribution of numerical estimation. The end-to-end comparison evaluates posterior accuracy on the same message streams and reporting checkpoints, with each system's call schedule specified in App. K. Text memory retains the reports while the explicit estimator stores their quantitative implications.

Ethics and reproducibility. The experiments use open-weight models, synthetic sensor messages, and a public geophysical series, with no human participants or sensitive records. Controlled monitoring tests do not certify a deployed system's safety. App. C.5 distinguishes registered criteria from later analyses; Apps. C.1, C.4, and B document prompts, checkpoints, decoding, validity, and pairing. App. K records the telemetry integrity audit.

B FIXED-HISTORY AND INFERENCE-PROTOCOL TESTS

B.1 PAIRED LOCAL-RESPONSE AUDIT

The Qwen3-8B fixed-history audit uses 30 trajectories and times $t \in \{5, 8, 11, 14\}$. A common numerical history is generated with sensor standard deviation 2, rounded to two decimal places, and then held fixed when the declared variance is changed to $R \in \{1, 4, 16\}$. At each R, the current reading is replaced with $z_t \pm \delta$ for $\delta \in \{1.5, 3, 6\}$. The short system prompt states $Q = 1$, the declared R, and prior $N(0, 4)$; the user prompt lists the fixed prefix and asks for a JSON posterior mean and interval. The current reading is displayed to eight decimal places. These $30 \times 4 \times 3 \times 3 \times 2 = 2{,}160$ calls all parse, giving $1{,}080$ paired contexts. Each exact K_t^* is computed for that time point under the declared variance and prior. In this linear-Gaussian model the gain depends on the variance schedule, not the observed numerical values.

For each paired context, Eq. 1 uses the two reported means. The signed span pairs the same trajectory, time, and perturbation scale at $R = 1$ and $R = 16$. Confidence intervals resample whole trajectories, preserving all within-trajectory pairs. Table 3 gives the full $R \times \delta$ measurements. At $\delta = 3$ the signed span is 0.021360 $[0.003126, 0.040065]$, divided by a reference span of 0.396623 to give GAR 0.053855 $[0.007882, 0.101015]$. The raw JSONL has SHA-256 prefix `7b4534af961c` and records checkpoint identity and decoding parameters for every call. The archived five-point slopes in App. D.2 use another prompt, independently drawn histories across noise levels, and a least-squares fit; they are a separate sensitivity protocol rather than a replication of these two-point differences.

Table 3: **Fixed-history local responses of Qwen3-8B.** Each cell has 120 paired contexts (30 trajectories, four time steps). K_{loc} changes only the current reading by $\pm\delta$ while holding the displayed prefix fixed. Intervals and the signed GAR use trajectory bootstrap; K_t^* is computed separately for each context under the declared variance. This protocol differs from the archived five-point local regression.

δ	Declared R	K_{loc} [95% CI]	Mean K_t^*	Signed GAR [95% CI]
1.5	1	$0.653389\ [0.589831, 0.713779]$	0.618056	$0.035085\ [-0.020100, 0.096765]$
	4	$0.647462\ [0.592066, 0.701387]$	0.391106	
	16	$0.639473\ [0.584556, 0.693228]$	0.221434	
3	1	$0.640465\ [0.601708, 0.678663]$	0.618056	$0.053855\ [0.007882, 0.101015]$
	4	$0.638173\ [0.601409, 0.671629]$	0.391106	
	16	$0.619104\ [0.581207, 0.656308]$	0.221434	
6	1	$0.691809\ [0.642110, 0.743098]$	0.618056	$0.113069\ [0.066992, 0.161470]$
	4	$0.685662\ [0.640185, 0.731881]$	0.391106	
	16	$0.646963\ [0.595919, 0.698445]$	0.221434	

B.2 Predictive-state and gain ladder

The staged Qwen3-8B test uses 30 trajectories, $R \in \{1, 16\}$, and the same four time points, yielding 240 contexts per condition. Every condition receives the complete history. A asks for the updated posterior mean, interval, and reported gain. B additionally supplies the exact predicted mean μ_t^- and variance P_t^- before the current reading. C also supplies the exact gain K_t^*, with the same predicted state. A separate arithmetic-only condition gives the numerical operands and asks for $P_t^-/(P_t^- + R)$ and $\mu_t^- + K_t^*(z_t - \mu_t^-)$. All 960 calls parse under reasoning-enabled decoding. Its pure arithmetic prompt is a different task from history-based state estimation.

For A, B, and C, raw mean error is $|\hat{\mu}_t - \mu_t^*|$, while normalized mean error is $|\hat{\mu}_t - \mu_t^*|/\sqrt{P_t^*}$. The raw means are 0.028688, 0.00007044, and 0.00001265, respectively; the normalized means are 0.0225874, 0.00007686, and 0.00000972. The arithmetic-only control has raw mean error 0.0036373 and normalized mean error 0.0046149. Paired normalized-error reductions are $0.0225105\ [0.0083767, 0.0411566]$ from A to B and $0.00006714\ [0.00000956, 0.00017825]$ from B to C. Every interval resamples trajectories. The arithmetic result does not by itself remove all arithmetic difficulty from a full-history query; it isolates a narrower computation with supplied operands.

Table 4: **Supplying the predictive state makes the update accurate.** Qwen3-8B under the reasoning-enabled decoding protocol; each condition has 240 valid queries on the same histories. Error is the absolute posterior-mean error divided by the reference posterior standard deviation. Brackets give 95% trajectory-bootstrap intervals. The arithmetic-only control is reported in App. B.2.

Condition	Information supplied before the update	Normalized mean error ↓
A	Observation history	$0.022587\ [0.008399, 0.041234]$
B	A + exact predicted mean and variance	$0.00007686\ [0.00001931, 0.00018761]$
C	B + exact Kalman gain	$0.00000972\ [0.00000820, 0.00001151]$

B.3 Reasoning-protocol boundary on held-out trajectories

The comparison reuses the exact prompts and paired readings of the base fixed-history run, matching message hashes row by row. Development trajectories are 0–7, holdout trajectories are 8–19, and trajectories 20–29 are reserved. The frozen gate required at least 95% valid development calls and a positive lower trajectory-bootstrap bound on paired absolute local-gain error reduction before evaluating the holdout. The gate passed: $576/576$ development and $864/864$ holdout calls parse. The development set contains 288 paired contexts and the holdout 432.

The base protocol disables thinking and uses temperature 0 with at most 512 generated tokens. Both protocols use top-$p = 0.95$, top-$k = 20$, and seed 20260920. The reasoning-enabled protocol enables thinking and uses temperature 0.6 with at most 32,768 tokens. The holdout reasoning run generated 6,969,394 output tokens over 864 calls, about 8,066 per call, compared with about 34 per call over the complete base fixed-history run. Thus this is a capability comparison between inference protocols

with different computational budgets, not a single-factor test of the thinking switch. The holdout absolute local-gain MAE is 0.300752 for base and 0.019444 for reasoning-enabled decoding; paired reduction is 0.281308 $[0.248492, 0.315204]$. The development results are in Table 5.

Joint response and posterior-mean accuracy. We also score each of these same 864 outputs against the exact posterior mean for its displayed, perturbed history. If μ_t^* denotes the reference at the center reading, the two targets are $\mu_t^* \pm \delta K_t^*$; recomputing the Kalman filter from the displayed histories verifies these targets. Table 5B reports both metrics by R. Posterior-mean MAE falls from 1.325353 to 0.055766, with paired reduction 1.269587 $[1.068770, 1.479765]$; RMSE falls from 2.008280 to 0.354772. Each of the 12 trajectories improves on both mean-error metrics. The 95th percentile of absolute posterior-mean error falls from 4.581 to 0.218. The new mean-error intervals use $50{,}000$ paired trajectory-bootstrap resamples (seed 20260926), retaining all times, variances, perturbations, and both query signs within each sampled trajectory. Thus the same-output comparison establishes improvements in both the reading response and the posterior level.

Table 5: **A reasoning protocol improves both local responses and posterior means.** A: local-gain MAE and paired trajectory-bootstrap reductions. B: joint errors on the same held-out outputs; each R has 288 calls (144 pairs). Prompts and histories match across protocols (App. B.3).

A. Development and held-out local-gain error

Split	Traj.	Valid calls	Base	Reasoning	Paired reduction [95% CI]
Development	8	$576/576$	0.295864	0.018579	0.277286 $[0.238649, 0.308463]$
Holdout	12	$864/864$	0.300752	0.019444	0.281308 $[0.248492, 0.315204]$

B. Same-output holdout evaluation: 12 *trajectories,* 864 *calls,* 432 *pairs*

	Local-gain MAE		Posterior-mean MAE		Posterior-mean RMSE	
R	Base	Reasoning	Base	Reasoning	Base	Reasoning
1	0.197	0.031	0.923	0.083	1.345	0.534
4	0.285	0.015	1.229	0.048	1.864	0.246
16	0.420	0.012	1.824	0.036	2.611	0.179
All	0.301	**0.019**	1.325	**0.056**	2.008	**0.355**

B.4 Multi-model fixed-history audit and number formatting

The $\delta = 3$ audit is repeated on Qwen3-4B-Instruct-2507 (FP8) and Mistral-7B-Instruct-v0.3 with the same 30 histories and four time points. Each model has 120 paired contexts per declared variance, or 360 pairs and 720 calls across $R \in \{1, 4, 16\}$. All $1{,}440$ calls across the two added models parse under greedy decoding. Each reference gain uses the declared variance and the corresponding time point; trajectory bootstrap resampling retains the cross-variance pairing.

Qwen3-4B's mean local responses are 0.584, 0.572, and 0.561, giving a signed span of 0.023 $[0.004, 0.043]$ and GAR 0.058 $[0.010, 0.108]$. Mistral-7B's responses are 0.682, 0.670, and 0.651, giving a span of 0.031 $[0.008, 0.055]$ and GAR 0.078 $[0.020, 0.139]$ (Table 1B). Together with Qwen3-8B, these results show small positive adaptation across the three checkpoints: only about 5–8% of the prescribed endpoint change. The absolute response levels differ across models.

A matched Qwen3-8B sensitivity check displays the perturbed current reading to two decimal places, matching the history, instead of eight. At the tested condition its local response is 0.635, compared with 0.638 at eight digits. This checks the local effect of display precision at that condition; the three-variance audit supplies the evidence about reliability adaptation.

B.5 Development budgets and reasoning traces

A development sweep varies the maximum generation budget $B \in \{512, 1024, 2048, 4096, 8192\}$ under thinking-enabled decoding at temperature 0.6. At $B = 512$, 64% of calls fail to parse after truncation. The reported local-gain MAEs at $B = 1024$, 2048, 4096, and 8192 are 0.184, 0.072, 0.022, and 0.019, respectively. Most of the measured improvement occurs by 4096 tokens. A local response requires both perturbed calls to parse. The budget summary does not yet specify complete-pair counts or the scoring of incomplete pairs, so these error values do not establish unconditional

completion performance. This development sweep is distinct from the held-out protocol above, whose maximum budget is 32,768 and whose mean output length is 8,066 tokens.

Manual inspection of successful traces finds explicit covariance prediction $P_t^- = P_{t-1} + Q$, gain calculation $K_t^* = P_t^-/(P_t^- + R)$, and innovation updates $\mu_t = \mu_t^- + K_t^*(z_t - \mu_t^-)$. A three-generation-seed follow-up reports mean held-out local-gain MAE 0.0194. These checks support the ability to carry out the recurrence under the reasoning protocol; the gain comparison remains the quantitative test of the resulting updates.

B.6 Discounted filtering and uncertainty scaling

A static discounted filter (Zhang et al., 2025) provides a structural comparison for the gain plateau. With fixed $R > 0$, discount factor $0 < \gamma < 1$, and no added process noise, its covariance recurrence is

$$P_t^- = P_{t-1}/\gamma, \qquad P_t = (\gamma/P_{t-1} + 1/R)^{-1}. \tag{4}$$

At the positive steady state, $1/P^*_{\text{disc}} = \gamma/P^*_{\text{disc}} + 1/R$, so

$$P^*_{\text{disc}} = (1-\gamma)R, \qquad K^*_{\text{disc}} = \frac{P^*_{\text{disc}}/\gamma}{P^*_{\text{disc}}/\gamma + R} = 1 - \gamma. \tag{5}$$

The gain is independent of R at this steady state. At finite time, the information recurrence instead gives $P_t^{-1} = \gamma^t/P_0 + (1-\gamma^t)/[(1-\gamma)R]$, retaining dependence on the initial variance. The derivation assumes $Q = 0$, whereas the tracking task has $Q = 1$.

The static rule fitted to telemetry oracle-checkpoint error selects $\gamma \approx 0.39$ on LLM-extracted inputs (Table 25). Its stylized steady-state gain $1 - \gamma \approx 0.61$ is close to Qwen3-8B's paired local response 0.62–0.64. This fit optimizes a numerical task baseline, rather than the model's reported trajectories; it does not identify an attention discount factor. Likewise, differences between pooled trajectory coefficients and local responses cannot by themselves be attributed to reporting-error attenuation. The paired contrast avoids the shared previous-report term and cancels common additive output offsets, while measuring a finite-range response.

Uncertainty supplies a separate comparison. Static discounting predicts a steady-state Gaussian interval width proportional to $\sqrt{R}$, hence a factor of 4 between $R = 1$ and 16. For the actual $Q = 1$ Kalman reference, the exact steady-state posterior variance and central 90% full width are

$$P^*_{\text{KF}}(R) = \frac{\sqrt{1+4R}-1}{2}, \qquad W_{90}(R) = 2\Phi^{-1}(0.95)\sqrt{P^*_{\text{KF}}(R)}. \tag{6}$$

Here Φ is the standard normal distribution function. At $R = 1, 4, 16$, the exact widths are 2.586, 4.111, and 6.182, giving an endpoint ratio of 2.390. The asymptotic law $W_{90} \propto R^{1/4}$ is not an exact finite-R ratio. Using the profile's prior $P_0 = 4$ and averaging steps 1–15 gives widths 2.619, 4.194, and 6.222, with ratio 2.376. Qwen3-8B's reported mean widths are 2.22, 3.12, and 3.98, expanding by 1.79. Thus, width adaptation is substantially smaller than both the exact-reference change and the fixed-discount steady-state prediction. Together with the outage and latent-state interval results in App. H, this limits a single static-discount account of both mean and uncertainty reporting.

B.7 Matched-extraction update engines

The update-engine comparison holds the extracted sensor-value stream fixed, separating numerical updating from language extraction. The telemetry evaluation uses 30 held-out trajectories with checkpoints at $t \in \{4, 8, 12\}$, giving 90 scored checkpoints; the public-series evaluation uses 38 held-out CO_2 windows. The supplementary inference runs use seed 20260920. The base updater receives the extracted readings and requests a posterior mean and interval from Qwen3-8B with greedy decoding and a 512-token limit. The reasoning updater uses the same extracted input with thinking enabled, temperature 0.6, and a 32,768-token limit. The reported calls for both LLM update engines parse successfully.

The code-tool updater generates a Python program implementing the Kalman recurrence with the declared dynamics and sensor variances, including $(Q, R_A, R_B) = (1, 1, 9)$ for telemetry. PDDA applies that recurrence directly, without generating an intermediate program. On telemetry, the base

and reasoning updater MAEs are 0.435 $[0.362, 0.518]$ and 0.038 $[0.024, 0.056]$; code-tool and PDDA are both 0.028 $[0.000, 0.061]$. The exact-update PDDA evaluation has median error 0; its mean retains the effects of the shared extracted inputs. The world-31 sensitivity analysis in App. K is a separate full-set evaluation.

The generated-token and execution-time summaries concern the update stage after extraction: approximately 42 tokens and 180 ms for the base updater, $7{,}850$ tokens and 14.2 s for reasoning, and 180 tokens and 450 ms for the code-tool route including local execution. Direct filtering uses no update-stage generation and executes in less than 1 ms. Language extraction remains a shared upstream cost. These measurements compare the numerical routes on their common extracted evidence; they do not make the full PDDA pipeline a zero-token system.

C Experimental protocol

C.1 Prompts and renderers

Numeric tracking. Each query specifies the generative model and asks for the current state from the complete observation history. Renderer A is used unless the condition explicitly changes the wording or reliability information. The baseline system prompt is

```
You are an expert state estimator tracking a robot on a 1D
track.  Physics:  each second the robot's position changes
by Gaussian noise with std 1.0 (pure random drift, no
velocity).  The sensor reads position with Gaussian noise
std 2.0.  Before any reading, the position is Gaussian:
mean 0.0, std 2.0.  Combine ALL readings optimally to
estimate the CURRENT position.
```

For each condition, the stated sensor standard deviation is $\sqrt{R}$. At time t, the user message contains one line for every observation available through that time. A fresh query receives this history rather than the model's previous answer, except in the explicit state-handoff experiment. Renderer B expresses the same numerical values with disjoint wording. The response instruction is

```
Reply with ONLY one JSON object:  {"mean":  <number>, "low":
<number>, "high":  <number>}
```

Here `low` and `high` specify a 90% credible interval. Missing-observation lines state that no reading was received. Prompt interventions change the noise description, add a block schedule, or repeat the sensor standard deviation on observation lines. The equation, example, and state-handoff conditions are specified in App. G.

Telemetry. Sensor A reports with probability 0.8 and variance $R_A = 1$, while sensor B reports with probability 0.6 and variance $R_B = 9$. Each trajectory lasts 12 time steps. Irrelevant messages are added on 25% of steps, and steps without reports use a designated no-message string. Message order is randomized within a step. Seven templates cover terse, verbose, approximate, hedged, indirect-reference, spelled-number, and correction-based reports. The common system prompt specifies the dynamics and noise levels.

```
You are monitoring a rover drifting on a 1D rail.  Physics:
each tick the position changes by Gaussian noise with std
1.0.  Two sensors exist:  sensor A reads position with noise
std 1.0; sensor B (informally 'the flaky one') with noise
std 3.0.  The good sensor means A. Before any reading,
position is Gaussian mean 0.0 std 2.0.  Messages may be
chatter with no reading; some spell numbers out; if a
message corrects itself, only the corrected value counts.
```

Table 6: **Reported cost of numerical updates on shared extractions.** Tokens and latency cover the updater stage only; the shared LLM extraction cost is excluded. PoT latency includes local code execution. Values are the reported mean output tokens and approximate single-update execution times.

Update engine	Generated tokens	Latency
Base LLM (single-pass)	~ 42	~ 180 ms
Reasoning LLM	$\sim 7{,}850$	$\sim 14{,}200$ ms
Code-tool (PoT Kalman)	~ 180	~ 450 ms
PDDA / Kalman	**0**	$< \mathbf{1}$ ms

The end-to-end model returns a mean and interval at steps 4, 8, and 12. The extraction model instead receives the following instruction at each time step.

```
For EACH message line below, output one JSON object per
line: {"sensor": "A"|"B", "value": <number>} for a real
reading, or {"sensor": null} for chatter/no reading.
Output nothing else.
```

Accepted sensor-value pairs are processed by the Kalman filter in message order using (Q, R_A, R_B). Checkpoint MAE averages errors at the three checkpoints. The registered alarm statistic compares one terminal decision per trajectory, using $P(x_{12} > 4) > \frac{1}{2}$. Under a Gaussian posterior this is equivalent to testing $\mu_{12} > 4$, so it is a mean-based decision check. The audited regex recognizes uppercase sensor names, `unit-A/B`, *the good sensor*, and *the flaky one*. It selects the last numerical span after a correction and excludes the digit in `CO2`. Numbers written in words remain outside its rules.

Public CO_2 evaluation. We use the weekly Mauna Loa series distributed with `statsmodels` (Seabold & Perktold, 2010; Keeling & Whorf, 2004). Its 2,284 weeks include 59 linearly interpolated missing values. The drift variance $\hat{Q} = 0.219$ is the sample variance of first differences on the first 60% of the series, ending on 1984-06-23. The candidate holdout extends from 1984-06-30 to 2001-12-29. Qwen3-8B uses 76 complete non-overlapping 12-week windows ending on 2001-12-15; Qwen3-4B uses the first 30, ending on 1991-05-18. Each window resets the prior to the training mean and variance, approximately 327.54 and 76.80, and is scored at weeks 4, 8, and 12. The observed series supplies the underlying path to the controlled message generator with $R_A = 1$ and $R_B = 9$. We report deviation from the oracle-parse Kalman mean and MAE to the observed series. The five registered point-estimate checks require deviation ratios at most 0.25 and 0.5 relative to end-to-end and regex, a series-MAE ratio at most 1.10 relative to oracle-parse filtering, and improvement over last-reading and running-mean baselines. Both Qwen checkpoints meet these checks. The oracle-parse filter is a model-based reference, not an optimal predictor of this public series.

C.2 Models and inference

The five-model profile contains Qwen3-1.7B, Qwen3-8B (Yang et al., 2025), Qwen3-4B-Instruct-2507 in its FP8 release, Mistral-7B-Instruct-v0.3 (Jiang et al., 2023), and OLMo-2-7B-Instruct (Team OLMo et al., 2025). Every 4B result refers to that FP8 checkpoint. The fixed-observation scale check additionally uses Qwen3-30B-A3B, with 3B active parameters. Local HuggingFace checkpoints are served through an OpenAI-compatible endpoint. The original base tracking experiments use greedy, non-thinking decoding; the new reasoning-protocol tests have their own settings in App. B.3. In base tracking, request errors are retried; parsing failures instead retain the previous mean, or the first observation at initialization. These fallback values remain in the recorded gain sequence. At $R = 16$, Qwen3-1.7B, 4B, and 8B have 14, 3, and 1 failures out of 600 queries, respectively. Interval and perturbation analyses use their own valid-output masks. The scale comparison in Table 14 uses Qwen3-30B-A3B.

C.3 Compute and infrastructure

Models run on a local GPU server with no training or finetuning. The original base behavioral experiments use greedy decoding, while the reasoning-protocol tests use the settings in App. B.3.

Behavioral queries use the local endpoint; steering uses batched generation with a default batch of 30. The two dense-sweep renderers require 7,200 queries in total. The recorded outputs and experiment configurations provide the sources for the reported measurements.

C.4 Pairing and seeds

The dense sweep and cross-trajectory factorial generate the latent path and standardized noise once per trajectory identifier, using SHA-256-derived seeds. With $x_{1:T}$ and $\varepsilon_{1:T} \sim \mathcal{N}(0,1)$ fixed, conditions use $z_t = x_t + \sqrt{R_t^{\text{cell}}}\,\varepsilon_t$. The within-trajectory factorial shares latent paths across all four conditions, but draws separate noise sequences for constant and switching actual-noise schedules. A/C share observations, as do B/D. Trajectory-level resampling retains the corresponding pairing. In the static-order control, $Q = 0$ and five permutations of the same ten observations preserve the exact posterior's sufficient statistics.

C.5 Provenance of reported experiments

Table 7 lists each experiment's role, registration status, and recorded outcome.

D The effective-gain estimator

D.1 Pooled regression and validation

Definition 1 pools updates from $t \geq 2$. It is descriptive because the previous model estimate appears in both the innovation and response. On the frozen exact-filter trajectories its difference from the steady-state reference is below 0.02 across the grid. For an unquantized constant-gain recurrence, substitution recovers the coefficient exactly when $\sum \nu_t^2 > 0$. For time-varying exact gains, the same estimator equals $\sum \nu_t^2 K_t / \sum \nu_t^2$, an innovation-energy-weighted average. Changing data noise can therefore change the regression weights as well as the responses.

Gain intervals resample whole trajectories. The five-model profile, renderer-A dense sweep, and steering gains use 600 resamples with seed 0; renderer B uses $1{,}000$ with seed 3. The stated-noise factorial and fixed-observation comparisons use $2{,}000$ with seed 7. Within-trajectory and exploratory GARs use $5{,}000$ with seed 42, and the constant-noise check uses $2{,}000$ with seed 42. HMM summaries use 800 trajectory resamples with seed 0. CO_2 intervals use $10{,}000$ window resamples, with seed 20260813 for posterior deviation and 20260814 for public-series MAE.

D.2 Local perturbation gains

A separate archived response measurement holds each observation prefix fixed and queries five values with evenly spaced offsets in $[-3, 3]$ around the current reading. A least-squares line gives a finite-perturbation slope when at least three responses parse. Each noise level uses 15 trajectories at steps 5, 8, 11, and 14, giving 60 contexts. Histories are independently sampled across R, and the longer robot prompt differs from the new short Bayesian prompt. Table 8 reports the slopes without the archived ratio that used only the $t = 5$ reference gain. The new paired, two-point protocol and its per-context normalization appear in App. B.1.

D.3 GAR normalization

Constant-noise GAR uses the steady-state reference in Prop. 1. Within-trajectory GAR uses the arithmetic mean reference gain in each phase on the included steps, following the declared schedule for prompt-only changes and actual noise for data switches. Replacing this denominator with the steady-state difference changes the B/C/D ratios by approximately 0.023, 0.049, and 0.032. We retain the phase-specific convention. Because pooled effective gain weights innovations by their squared magnitude, GAR equal to one denotes the prescribed phase-mean change rather than an exact self-score identity for a time-varying Kalman filter.

Table 7: Provenance of the reported experiments. Original registrations, later code-specified analyses, and exploratory controls are distinguished. The new fixed-history and reasoning-protocol measurements were specified after the original registration.

Experiment	Role in paper	Registration	Outcome
Single-R profile, 5 models (§3)	confirmatory	preregistered	pass
Outage, static order, and noise endpoints	confirmatory	preregistered	pass
HMM analogue (§3)	confirmatory	preregistered	pass
Dense 8-point sweep, renderer A	confirmatory	preregistered (span ≤ 0.12)	span 0.064
Renderer-B rephrasing (gain + fusion)	confirmatory	preregistered, SHA-256-frozen	span 0.068, fusion pass
Telemetry + audited evaluator	confirmatory	criteria frozen; evaluator audited	pass
Public CO_2 anchor (8B, 4B)	confirmatory	preregistered (5 checks)	all pass
Mistral runtime transfer	confirmatory gate	preregistered	**fail (reported)**
Thinking-mode control	control	preregistered unlock criterion	inconclusive (App. M)
Stated $\times$ experienced 2×2	follow-up analysis	design fixed in code, post-registration	§4
Stated-R ladder (8B, 30B-A3B)	follow-up analysis	design fixed in code, post-registration	§4
Within-trajectory factorial (B/C/D)	follow-up analysis	design fixed in code, post-registration	§4
Local perturbation gain	robustness	design fixed in code, post-registration	App. D.2
Fixed-history 8B paired audit	primary follow-up	code-specified, post-registration	$2{,}160/2{,}160$ valid
Multi-model paired audit (4B, Mistral)	replication follow-up	code-specified, post-registration	$1{,}440/1{,}440$ valid
Predictive-state and gain ladder	diagnostic follow-up	code-specified, post-registration	$960/960$ valid
Reasoning-protocol boundary	capability follow-up	frozen dev/holdout gate	$1{,}440/1{,}440$ valid
Development token budgets	capability follow-up	post-registration	App. B.5
Prompt ladder (formula/example/state)	robustness	design fixed in code, post-registration	App. G
Reference-interval comparison	descriptive	recomputed from frozen trajectories	App. H
Residual-stream probe (S1/S2 splits)	confirmatory	preregistered, frozen criteria	S1 pass; S2 below criterion (App. I)
Filter-calibrated steering (20 arms)	confirmatory	preregistered incl. outcome branches; errata recorded pre-run	readout changes; gain not restored
Static-discount derivation	theoretical comparison	post-registration	App. B.6
Matched update engines	systems ablation	code-specified, post-registration	App. B.7
Latent-state interval coverage	calibration audit	post-registration	App. H
Telemetry sensitivity	robustness audit	code-specified, post-registration	App. K

E FULL GAIN RESULTS

Table 10 reports the three-point gains and trajectory counts for all five models. Table 11 gives Qwen3-8B's eight-point sweep under both renderers. Table 12 compares model estimates with simple rules on the same observations. The separately fitted constant gains are 0.66, 0.42, and 0.22 at $R = 1$, 4, and 16. Those descriptive baseline rows fit on half the trajectories and score the full set, which includes the fitting half. They are not held-out estimates of a learned rule's generalization. The

Table 8: **Archived five-point perturbation slopes.** Each declared noise level uses 15 trajectories at four times, giving 60 contexts. A within-context five-point regression estimates the slope. Histories are independently drawn across declared noise levels, so these rows are not paired across R and differ from Table 3 in prompt and estimand.

	R=1	R=4	R=16
K_{loc}	0.40	0.38	0.31

distinct system comparison in Table 21 reports held-out results for all fitted update rules.

The longer-history results help characterize the fitted response. At $R = 4$, the coefficient decreases from approximately 0.64 at $T = 5$ to 0.43 at $T = 15$ through 30. Thus, the gain level can change with history length even when its dependence on reliability remains weak. The Qwen and Mistral trajectory-fit coefficients generally fall below the low-noise reference and above the high-noise reference. OLMo differs in level, remaining at or below the reference on the tested grid and yielding the weakest one-coefficient fit, with R^2 from 0.42 to 0.51. Its endpoint gain change is nevertheless much smaller than the required 0.40.

F Channel factorial details

Stated and experienced noise. The factorial in Table 13 crosses $R_{\text{prompt}}, R_{\text{data}} \in \{1, 16\}$ on shared latent paths and standardized noise. Each cell has 40 Qwen3-8B trajectories or 30 Mistral-7B trajectories, with 15 steps per trajectory. Stated-noise GAR is 0.007 and 0.025, respectively; experienced-noise GAR is 0.077 and 0.095. Absolute gain levels differ between this shared-trajectory cohort and the independently sampled profile: the stated=1/data=1 cells are 0.433 for Qwen3-8B and 0.556 for Mistral, compared with 0.49 and 0.65 in Table 9. The reliability contrasts are computed within their respective cohorts. The fixed-observation comparison in Table 14 holds one $R = 4$ observation history fixed for each trajectory and changes only the stated standard deviation over $\{0.5, 1, 2, 4, \sqrt{32}\}$. This covers a 128-fold range of stated variance without changing a single observation.

Within-trajectory factorial. The four conditions share 30 latent trajectories of $T = 20$ steps. Switching schedules balance low-noise-first and high-noise-first orders with block lengths 4, 5, or 6. A uses constant actual noise $R = 4$ and a constant description. B switches actual noise between 1 and 16 but retains the constant description. C keeps actual noise at 4 while declaring a switching schedule and marking the standard deviation on each observation line. D switches actual noise and uses the same schedule-plus-line presentation. A/C share observations, and B/D share observations. Table 15 gives phase-specific gains and normalized contrasts. Figure 2 shows the sampling distributions from 2,000 paired trajectory-bootstrap draws for the stated-noise contrast (seed 7) and 5,000 draws per schedule contrast (seed 42). Violin widths use a Gaussian kernel with Scott bandwidth and are normalized within each condition; dots and bars retain the pooled estimates and percentile 95% intervals. These are bootstrap sampling distributions, not distributions of independent trajectory effects.

Exploratory order comparison. The alternate-prompt exploratory condition in Table 15 gives GAR 0.654 $[0.51, 0.79]$ from an unpaired cohort with low noise presented first. Its low- and high-noise phase gains are 0.494 and 0.173. This design aligns lower noise with earlier history positions and higher noise with later positions. The history-length dependence in App. E makes that ordering a potential confound. In the counterbalanced factorial, declared-only condition C gives GAR 0.205 and jointly switched condition D gives 0.134. These smaller effects support partial adaptation under balanced schedules. Because the exploratory comparison also changes prompt and cohort, it does not isolate the contribution of order alone.

G Archived formula-prompt controls

The equation condition supplies the Kalman gain, mean update, and variance recurrence; the example condition adds two numerical steps. The assisted one-step condition feeds back the model's

Table 9: **Five-model state-estimation profile.** The correlation r uses posterior means at $R = 4$. Effective gains are measured at matched stated and actual noise, and $\Delta\hat{K} = \hat{K}_1 - \hat{K}_{16}$ is compared with the required 0.40. The R^2 range describes the fit across the three noise levels. Outage is the interval-width ratio over a three-step gap. Order is terminal-mean spread in posterior standard deviations in the separate static test with $Q = 0$. Table 10 gives gain confidence intervals and trajectory counts.

Model	r	$\hat{K}_1$	$\hat{K}_4$	$\hat{K}_{16}$	$\Delta\hat{K}$	R^2 (range)	outage	order
Qwen3-1.7B	0.962	0.51	0.55	0.58	-0.07	0.72 to 0.76	1.46	1.35
Qwen3-4B	0.975	0.50	0.41	0.47	+0.03	0.72 to 0.78	1.04	0.88
Qwen3-8B	0.977	0.49	0.43	0.45	+0.04	0.73 to 0.77	1.07	0.68
Mistral-7B	0.972	0.65	0.56	0.55	+0.10	0.79 to 0.83	1.13	1.29
OLMo-2-7B	0.927	0.25	0.26	0.22	+0.03	0.42 to 0.51	1.09	0.65
Kalman filter	1.000	0.62	0.39	0.22	+0.40	1.00	1.71	0.00

Table 10: Effective gains, trajectory counts, and trajectory-bootstrap 95% confidence intervals for the three-noise-level comparison in Table 9.

Model	$N_{1,16}$ / N_4	R=1	R=4	R=16
Qwen3-1.7B	40 / 60	0.51 [0.46,0.55]	0.55 [0.51,0.58]	0.58 [0.54,0.61]
Qwen3-4B	40 / 60	0.50 [0.46,0.53]	0.41 [0.39,0.43]	0.47 [0.43,0.51]
Qwen3-8B	40 / 60	0.49 [0.45,0.53]	0.43 [0.41,0.46]	0.45 [0.40,0.49]
Mistral-7B	40 / 60	0.65 [0.61,0.68]	0.56 [0.53,0.60]	0.55 [0.50,0.59]
OLMo-2-7B	40 / 60	0.25 [0.22,0.29]	0.26 [0.22,0.31]	0.22 [0.19,0.25]
Kalman	n/a	0.62	0.39	0.22

Table 11: Qwen3-8B's paired eight-point sweep with two observation renderers. Each point uses 30 trajectories and 420 eligible updates, after omitting the first step. Wording changes both level and direction: renderer B has signed endpoint span -0.050 (GAR ≈ -0.075). Maximum-minus-minimum spans are 0.064 for A and 0.068 for B, versus the reference span 0.67.

R	0.25	0.5	1	2	4	8	16	32
K^*	0.83	0.73	0.62	0.50	0.39	0.30	0.22	0.16
$\hat{K}$ (renderer A)	0.48	0.47	0.46	0.45	0.45	0.43	0.43	0.42
$\hat{K}$ (renderer B)	0.65	0.67	0.68	0.66	0.70	0.72	0.72	0.70

Table 12: RMSE to the exact posterior mean on each model's own observation sequences. Simple estimators reuse those observations. The fitted-gain baseline is trained on half the trajectories and evaluated on the full set, including its fitting data, so it is a descriptive comparison rather than a held-out generalization result. Bold marks the best end-to-end model.

Estimator	R=1	R=4	R=16
oracle-parsed simple estimators			
last observation	0.62	1.56	3.50
running mean	1.36	1.18	1.53
fitted gain	0.13	0.17	0.04
end-to-end models			
Qwen3-1.7B	0.73	1.00	2.31
Qwen3-4B	**0.54**	0.74	2.19
Qwen3-8B	0.57	**0.71**	1.77
Mistral-7B	0.56	1.01	2.28
OLMo-2-7B	1.04	1.26	**1.43**

previous mean and reference covariance. The parser does not retain the requested gain field, so the implementation uses reference covariance rather than a model-updated covariance. It does not hand over an oracle mean. In this 20-trajectory, 10-step experiment, baseline gains are 0.503 and 0.456 at

Table 13: **Separating stated and actual noise.** Cells share latent trajectories and standardized noise draws. Within each row, changing the stated noise leaves the observations fixed; between rows, the same noise draw is rescaled. The reference row gives Kalman gains under the stated sensor model. Brackets show trajectory-bootstrap 95% confidence intervals.

		stated R=1	stated R=16
Qwen3-8B	data R=1	0.433 [0.40,0.46]	0.433 [0.40,0.47]
	data R=16	0.406 [0.38,0.44]	0.400 [0.37,0.43]
Mistral-7B	data R=1	0.556 [0.50,0.61]	0.544 [0.49,0.60]
	data R=16	0.516 [0.48,0.55]	0.508 [0.47,0.54]
Kalman filter		0.618	0.221

Table 14: Fixed-observation comparison with five stated noise levels. Each trajectory retains one $R = 4$ observation history while the stated variance changes over a 128-fold range. Gain spans are 0.033 for Qwen3-8B and 0.033 for Qwen3-30B-A3B. Brackets show trajectory-bootstrap 95% confidence intervals.

stated R	0.25	1	4	16	32
K^*(stated)	0.83	0.62	0.39	0.22	0.16
Qwen3-8B	0.46 [0.42,0.49]	0.42 [0.39,0.45]	0.44 [0.41,0.47]	0.43 [0.40,0.45]	0.43 [0.40,0.45]
Qwen3-30B-A3B	0.44 [0.41,0.48]	0.41 [0.38,0.44]	0.42 [0.38,0.45]	0.42 [0.39,0.45]	0.41 [0.38,0.45]

Table 15: Within-trajectory reliability conditions. The paired factorial design counterbalances block order. Both C and D provide a switching schedule and per-line standard deviations. The exploratory condition uses separate unpaired, low-noise-first trajectories, with different wording and priors. GAR uses the difference in mean reference gains over the same phases and eligible steps.

Design	Channel	$\hat{K}_{\text{low}}$	$\hat{K}_{\text{high}}$	K^*_{low}	K^*_{high}	GAR [95% CI]
factorial (B)	experienced only	0.459	0.508	0.659	0.167	-0.098 [-0.23, 0.04]
factorial (C)	declared only	0.354	0.253	0.659	0.167	0.205 [0.03, 0.38]
factorial (D)	both	0.312	0.246	0.659	0.167	0.134 [-0.03, 0.31]
exploratory	both, alternate prompt	0.494	0.173	0.646	0.154	0.654 [0.51, 0.79]

Table 16: **Explicit instructions and assisted updates.** Gains and differences from K^*; cohorts give trajectories N and steps T. The 20×10 cohort's own baseline is 0.503 at $R = 1$ and 0.456 at $R = 16$. Assistance supplies the previous model mean and reference covariance.

Prompt	Cohort ($N \times T$)	$\hat{K}_1$	Δ_1	$\hat{K}_{16}$	Δ_{16}
Kalman filter	—	0.618	0.000	0.221	0.000
baseline	40×15	0.489	-0.129	0.447	+0.227
+ noise warning	40×15	0.504	-0.114	0.500	+0.279
+ reasoning scaffold	40×15	0.546	-0.072	0.484	+0.263
+ Kalman formula	20×10	0.544	-0.074	0.533	+0.312
+ worked example	20×10	0.507	-0.111	0.503	+0.283
+ assisted update	20×10	0.471	-0.147	0.469	+0.248

$R = 1$ and $R = 16$. Adding equations raises the high-noise gain to 0.533, away from 0.221. Table 16 distinguishes this baseline from the 40-trajectory, 15-step prompt controls.

H Reference-interval agreement and uncertainty dynamics

Table 17 reports the fraction of 90% intervals containing the reference posterior mean and the Winkler interval score evaluated at that mean (Gneiting & Raftery, 2007). The recorded scoring target is the reference mean, not the latent state, so these values measure agreement with the reference estimate rather than state coverage or calibration. Interval-width dynamics provide a separate uncertainty

Table 17: **Reported intervals evaluated against the reference posterior mean.** Inclusion is the fraction containing that mean, and IS is the interval score at the same target. Empirical latent-state coverage is reported separately in Table 18.

Model	R=1		R=16		Outage
	Inclusion	IS	Inclusion	IS	Inclusion
Qwen3-1.7B	0.88	2.77	0.50	19.06	0.79
Qwen3-4B	0.98	3.10	0.85	11.13	1.00
Qwen3-8B	0.95	2.68	0.86	8.44	0.95
Mistral-7B	0.98	3.38	0.85	11.26	0.88
OLMo-2-7B	0.78	5.22	0.84	8.47	0.80

check. Across a three-step gap, widths grow by 1.04 to 1.46, compared with 1.71 for the reference. In the static-order test, the standard deviation of terminal means across five permutations is 0.65 to 1.35 reference posterior standard deviations, although the exact posterior is order-invariant.

Where the simulated latent x_t was saved, we separately score empirical state coverage (Table 18). For Qwen3-8B, reported 90% intervals cover the latent state in 0.725 of valid cases at $R = 1$ and 0.614 at $R = 16$; during the 120 gap steps, coverage is 0.533. The $R = 1$ and $R = 16$ trajectory-bootstrap intervals are $[0.673, 0.770]$ and $[0.564, 0.663]$. The original $R = 4$ profile did not retain latent states. Its exact-posterior probability mass inside a reported interval is a conditional expected coverage quantity, not a realized empirical coverage observation.

I Reliability probes and steering inside the model

Internal experiments use Qwen3-8B in bf16 with greedy, non-thinking decoding and the same chat template as the dense sweep. Contexts are reconstructed from the frozen trajectory archive. Probe splits and steering decision criteria were fixed before their corresponding evaluations. The diagnostic corrections described below were recorded before the associated confirmatory reruns.

I.1 Activation collection and probes

The context set contains 8 noise levels, 30 trajectories, and 15 steps, giving $3{,}600$ queries. Reconstruction matches the archived mean field in 9/10 sampled contexts. The archive stores 37 hidden states at the final prompt token, comprising the embedding output and successive layer states with the last state taken after final normalization. Probes predict $\log_2 R$, per-step K_t^*, $\log P_t$, and Δ_t/ν_t; the last target uses only $|\nu_t| > 0.3$. Noise descriptions and generated observations co-vary, so the noise probe decodes the combined condition. Regularization is selected by trajectory cross-validation within the training data.

Split S1 holds out trajectories within each noise level. It gives $R^2 = 0.978$ at layer 5, with 0.00 for shuffled labels and an embedding-level control near chance. Split S2 instead holds out entire noise levels, training on $\{0.25, 1, 2, 16, 32\}$ and testing on $\{0.5, 4, 8\}$. The best S2 score is 0.42 at layer 28, below its preregistered criterion of 0.5; early-layer transfer scores are nonpositive. The next analysis examines how numerical format contributes to this difference in generalization.

I.2 Numerical format and reliability geometry

The eight noise-level centroids separate into two groups associated with the written standard deviation. Short numbers $0.5, 1, 2, 4$ correspond to variances $\{0.25, 1, 4, 16\}$, while long decimals such as 0.707107 correspond to $\{0.5, 2, 8, 32\}$. The first principal component explains 63% of the cross-level variance and largely separates these formats. Magnitude is ordered within each group, with all four levels in order. Directions are stable across two trajectory split schemes, with cosine similarities from 0.90 to 0.96, but the two within-group magnitude directions have cosine about 0.3. Thus, one global readout direction need not follow the local direction of changing magnitude.

Within-format comparisons restore the ordering at $R = 0.5$ that is obscured by the global readout. In a leave-one-level-out interpolation diagnostic, separate fold-specific constant midpoint predictions can produce a pooled $R^2 = 1$ because the target level changes between folds. A single global constant

Table 18: **Empirical latent-state coverage of reported 90% intervals.** The score tests whether the saved latent x_t lies inside each valid reported interval, rather than whether the reference posterior mean does. The latter is a different quantity reported in Table 17. n gives valid intervals.

	$R = 1$		$R = 16$	
Model	n	Coverage	n	Coverage
Qwen3-1.7B	599	0.591	586	0.411
Qwen3-4B	600	0.848	597	0.710
Qwen3-8B	600	0.725	599	0.614
Mistral-7B	600	0.830	600	0.733
OLMo-2-7B	600	0.648	600	0.685

cannot do so. We therefore report per-level bias, whose maximum is 0.76 in $\log_2 R$ units; late-layer interpolation preserves order in all six cases. Rank correlations use tie-aware computation. The global ridge direction has cosine about 0.12 with centroid chords, motivating within-format centroid differences for steering.

I.3 Filter-calibrated steering

Steering vectors use training-trajectory centroids at layer 28, where S2 transfer is highest. Write $\mu(R)$ for the activation centroid at noise level R, distinct from the scalar state estimate. The down direction is $\mu(0.25) - \mu(4)$, within one numerical format. The up direction is $1.5[\mu(16) - \mu(4)]$, extrapolating that format's magnitude direction by three $\log_2 R$ units. The cross direction $\mu(32) - \mu(4)$ changes format as well and is exploratory. A fixed random direction is norm-matched to the up direction at each dose. The format control is the long-minus-short centroid difference after removing its component along the shared magnitude axis.

At dose $g \in \{1, 2, 4, 8\}$, the intervention adds $g\Delta$ to zero-indexed block 27 at the last prompt position and every generated position. Within-run hook checks recover the intended state and reproduce hook-off tokens at zero dose on three contexts with a 64-token budget. The closest archived state is separated from adjacent layers by a 29-fold margin, using a relative tolerance for bf16 differences. Formal arms use batched generation and a 220-token budget. The zero-dose check validates the local hook, not exact equivalence of the full intervention run to the earlier archive.

The registered directional outcome requires both within-format vectors to move gains toward their targets, intervals excluding zero, and control effects below half the directional effect. The alternative branch additionally required inert controls. Some control intervals exclude zero, so the observed pattern is not claimed to satisfy that branch unchanged. The supported result is that frozen probes at layer 32 and final normalization shift substantially, while all gain changes remain within 0.019 of the archived baseline. This is far below the required adjustment; small cross-run differences do not establish an exact zero effect.

J Discrete-state comparison

Qwen3-8B estimates a state in a line of five rooms. At each step, the state remains in place with probability 0.5 or moves left or right with probability 0.25, reflecting at the endpoints. The sensor reports a room with accuracy a, and the model returns probabilities over rooms, which are normalized before scoring. For each accuracy, we divide the model's mean ℓ_1 belief movement by the exact forward posterior's mean movement on the corresponding trajectories. The exact-reference ratio is therefore one in either condition. The observed ratios are 0.435 for $a = 0.9$ and 0.556 for $a = 0.6$, indicating greater relative under-updating in the former. Their difference does not determine the ordering of unnormalized movement because the denominators differ. The forward algorithm is checked against exhaustive path enumeration.

K Telemetry, PDDA, and transfer details

Comparison unit and calls. Both systems receive the same message stream and known dynamics. PDDA extracts the current step's messages at each of the 12 time steps and maintains the numerical

Table 19: **Qwen3-8B steering at declared $R = 4$.** Every condition uses layer-28 interventions, 30 trajectories, and 15 steps. Effective gains follow Def. 1. The paired difference Δ is measured from the unsteered gain 0.448, with trajectory-bootstrap 95% confidence intervals. The final columns report shifts in frozen probe outputs at layer 32 and after final normalization, in $\log_2 R$ units. Parse failure is 0.000 in every condition. Target restoration requires gain changes $+0.38$ or -0.29 for the two within-format directions.

arm	g	$\hat{K}$	Δ vs. base	95% CI(Δ)	dec. $\Delta \log_2 R$ (L32)	(L36)
toward reliable	1	0.449	+0.001	[-0.001, +0.004]	-3.02	-2.14
	2	0.444	-0.004	[-0.009, +0.000]	-6.19	-4.30
	4	0.444	-0.004	[-0.008, +0.000]	-12.37	-8.65
	8	0.454	+0.006	[-0.002, +0.017]	-25.03	-17.12
toward noisy	1	0.444	-0.003	[-0.008, -0.000]	+2.32	+1.75
	2	0.444	-0.004	[-0.009, -0.000]	+4.67	+3.51
	4	0.440	-0.008	[-0.015, -0.001]	+9.19	+6.93
	8	0.436	-0.011	[-0.020, -0.004]	+17.86	+13.83
cross-format chord	1	0.445	-0.003	[-0.007, +0.001]	+2.79	+2.23
	2	0.443	-0.005	[-0.010, -0.000]	+5.45	+4.47
	4	0.440	-0.007	[-0.013, -0.002]	+11.01	+9.13
	8	0.428	-0.019	[-0.031, -0.008]	+21.68	+18.35
random direction	1	0.440	-0.007	[-0.013, -0.003]	-0.01	+0.01
	2	0.442	-0.005	[-0.010, -0.002]	-0.02	+0.11
	4	0.445	-0.003	[-0.009, +0.002]	-0.25	+0.14
	8	0.434	-0.013	[-0.023, -0.004]	-0.53	+0.48
format axis	1	0.445	-0.003	[-0.008, +0.002]	-0.06	-0.08
	2	0.437	-0.010	[-0.023, -0.002]	+0.03	+0.03
	4	0.436	-0.012	[-0.023, -0.003]	-0.10	+0.06
	8	0.429	-0.018	[-0.032, -0.007]	-0.21	+0.71

Table 20: Qwen3-8B on a five-state HMM, with 50 trajectories of 10 steps per sensor accuracy. TV measures total variation from the forward posterior. The update ratio is model mean belief movement divided by exact mean movement at that accuracy. The reference ratio is one in both conditions. A smaller ratio for the accurate sensor indicates greater relative under-updating, not necessarily smaller absolute movement. The $\pm$ terms give half-widths of bootstrap 95% confidence intervals.

Sensor accuracy	TV	argmax agree	update ratio
$a = 0.9$	0.468 ± 0.031	0.658 ± 0.043	0.435 ± 0.048
$a = 0.6$	0.348 ± 0.019	0.534 ± 0.050	0.556 ± 0.048

posterior between calls. End-to-end estimation queries the accumulated reports at the three checkpoints 4, 8, and 12. All posterior errors are evaluated at those same checkpoints. Thus the comparison matches evidence and output times, while the call schedules and context lengths differ.

Full results and transfer criteria. Table 21 reports the telemetry and public-series measurements, including Qwen3-4B, omitted and spurious extractions, and reference rules supplied with true readings. Table 22 keeps the original decision criteria and outcomes for message rephrasing and model transfer. The Mistral criterion is not met.

Representative messages. Table 23 shows recorded messages and extractor outputs. The examples illustrate numbers written in words and the indirect name *the flaky one*, which the audited regex can resolve.

Extraction evaluation. The telemetry set contains $1{,}029$ stored sensor readings. A match requires the correct sensor and an absolute value error below 0.01. Unmatched-reference counts include numerical errors as well as absent extractions; unmatched outputs are counted separately. The audited regex has recall 0.866, and recorded LLM extraction has recall 0.989 with 0.3% unmatched outputs. The same regex implementation supplies the statistics and trajectories. Earlier regex outputs with recall 0.804 remain archived. A template-aware parser reaches 0.999. The frozen matching and

Algorithm 1: Perception-Dynamics Decoupled Agent (PDDA). The model extracts observations, the filter maintains (μ, P), and the decision rule evaluates the resulting belief.

Input: message stream $m_{1:T}$; declared dynamics Q; declared sensor variances $\{R_s\}$; decision rule g
initialize exact filter $(\mu, P) \leftarrow (\mu_0, P_0)$;
for *time step* $t = 1, \ldots, T$ **do**
  $(\mu, P) \leftarrow (\mu,\ P + Q)$ `// Layer 2 predict;`
  $J_t \leftarrow$ LLM-EXTRACT(m_t) `// Layer 1 outputs {sensor s, value v} or abstention;`
  for $(s, v) \in J_t$ *in message order* **do**
    $K \leftarrow P/(P + R_s)$; $\mu \leftarrow \mu + K(v - \mu)$; $P \leftarrow (1 - K)P$;
  report $\mathcal{N}(\mu, P)$ at checkpoints and evaluate $g(\mu, P)$ at the terminal step;

Table 21: **Full telemetry and public-series results.** Telemetry error averages three checkpoints per trajectory. Omission and spurious-reading rates describe the extractor. Public-series results use 76 Qwen3-8B windows and 30 Qwen3-4B windows. The dagger marks true sensor readings; the double dagger marks fitted rules evaluated only on the held-out half. Reference rows give the oracle-parse filter on that half and on the full set. Dynamic discounting with $\gamma^* = 1$ recovers Kalman estimation, so differences from full-set PDDA reflect the evaluation subset. Public-series confidence intervals appear in Table 27.

		Language telemetry				CO_2 8B		CO_2 4B	
Extraction	Update	MAE	med.	omit	extra	dev.	MAE	dev.	MAE
none (end-to-end agent)	LLM	0.471	0.301	n/a	n/a	0.398	0.723	0.451	0.699
regex	Kalman	0.211	0.023	13.4%	0.0%	0.170	0.689	0.118	0.659
LLM (PDDA)	Kalman	**0.017**	**0.000**	**1.1%**	**0.3%**	**0.032**	0.605	**0.045**	0.633
oracle†	last reading	1.743	1.103	n/a	n/a	1.840	1.908	1.787	1.840
oracle†	running mean	1.144	0.936	n/a	n/a	0.748	1.135	0.774	1.150
LLM	fitted gain‡	0.736	0.556	n/a	n/a	0.960	1.128	1.033	1.138
LLM	two-gain‡	0.120	0.070	n/a	n/a	0.592	0.837	0.554	0.825
LLM	Zhang static‡	0.091	0.032	n/a	n/a	0.090	0.597	0.111	0.700
LLM	Zhang dynamic‡ (γ^*=1)	0.028	0.000	n/a	n/a	0.038	0.617	0.055	0.707
oracle†	*Kalman, same half*	0.000	0.000	n/a	n/a	0.000	0.627	0.000	0.656
oracle†	*Kalman (reference, full set)*	0.000	0.000	n/a	n/a	0.000	0.609	0.000	0.609

Table 22: **Model and wording transfer.** Renderer B regenerates messages from stored readings with disjoint vocabulary, also resolving one numerical inconsistency in the original renderer. The table reports Qwen3-8B and Mistral-7B separately against the registered thresholds.

Test	Quantity	Criterion	Value	Verdict
Renderer-B (Qwen3-8B)	gain span, 8-point sweep	≤ 0.12	0.068	pass
	fusion dev. ratio to e2e	≤ 0.25	0.0002	pass
	fusion dev. ratio to regex	≤ 0.50	0.0010	pass
Mistral-7B runtime	fusion dev. ratio to e2e	≤ 0.25	0.311	**fail**
	fusion dev. ratio to regex	≤ 0.50	0.841	**fail**

downstream scores are retained, with the following integrity check separating reference errors from extractor errors.

Message integrity and paired sensitivity. One of the 138 spelled-number messages loses a tens digit. In world 31 at tick 8, the report says *minus two point three*, while the stored reading is -12.3. The model extracts -2.3, correctly following the visible message. This discrepancy contributes 52.20% of the full-set PDDA absolute error and must not be attributed to the extractor. Table 24 retains all archived results and separately excludes this entire world from both systems. The exclusion was introduced after the audit, not preregistered, and leaves 59 paired worlds with 177 checkpoints. It does not alter stored readings or outputs. With both systems paired by world and checkpoint, a 100,000-resample trajectory bootstrap gives an end-to-end-minus-PDDA MAE reduction of 0.4543

Table 23: Recorded telemetry messages and corresponding extractions. Both the audited regex and the LLM resolve the indirect sensor name in the second row. Numbers written in words in the first row require a capability absent from the regex rules. Full-set recall is 0.866 for regex and 0.989 for the LLM; a parser with access to the seven generation templates reaches 0.999.

Operator message	regex	LLM
`A log entry - value approx five point two`	n/a	A=5.2
`the flaky one came back with 2.61`	B=2.61	B=2.61
`B says it's around 6.7, take it with a grain of salt`	B=6.7	B=6.7

Table 24: Post-audit sensitivity analysis, not preregistered. Both systems exclude the entire world 31, whose visible message contradicts its gold value, and use the same remaining 59 worlds. MAE is relative to the archived oracle posterior mean at steps 4, 8, and 12. The ratio is end-to-end MAE / PDDA MAE. The full set retains the known defect; no messages, gold readings, or predictions are regenerated or repaired.

Sample	Worlds	Checkpoints	E2E MAE	PDDA MAE	Ratio
All archived worlds	60	180	0.470944	0.016637	28.306
Exclude world 31	59	177	0.468425	0.008087	57.924

$[0.3862, 0.5285]$ on all 60 worlds. The post-audit 59-world sensitivity gives 0.4603 $[0.3921, 0.5344]$ under the same resampling rule.

Fitted update rules. Task baselines from the related discounted-filter families (Zhang et al., 2025) are fit to oracle-checkpoint MAE on half the data and evaluated on the other half. This objective differs from fitting a behavioral account of LLM belief trajectories. The split uses 30 telemetry trajectories and 38 or 15 public-series windows. Comparisons among fitted rules and LLM-extracted Kalman filtering hold the extracted stream fixed; oracle-parse rows instead provide a distinct extraction reference.

The fitted static rule selects $\gamma^* = 0.3863$ on LLM-extracted telemetry inputs and 0.3830 on oracle inputs, rounded to 0.39 and 0.38, respectively. The structural steady-state comparison is derived in App. B.6. The dynamic family selects $\gamma^* = 1.00$, yielding held-out telemetry MAE 0.028. On the public series, the static rule selects $\gamma^* = 0.66$ and gives 0.597. Kalman filtering on the same LLM extractions gives 0.617, compared with 0.627 for oracle extractions. These are held-out window results. The full-set values 0.605 and 0.609 use all 76 windows and have a different denominator.

Sensor-model misspecification. Table 26 varies the assumed variance $\hat{R}_B$ while keeping the true variance at 9. Underestimating it by a factor of three gives MAE 0.265, while overestimating it by the same factor gives 0.114. Both are below the end-to-end value 0.471, but they have different effects. The ninefold overestimate gives 0.151 and is reported as a separate condition.

Public-series confidence intervals. Table 27 resamples matching complete windows for paired systems. Windows are non-overlapping but may remain temporally dependent; these intervals summarize the window-sampling procedure, not uncertainty across sites or arbitrary future periods.

L PROOFS

Proof of Proposition 1. Let the prediction variance be $\bar{P} = P + Q$. With the update $P = (1-K)\bar{P}$ and $K = \bar{P}/(\bar{P}+R)$, the steady state satisfies $\bar{P} = \frac{\bar{P}R}{\bar{P}+R} + Q$. Equivalently, $\bar{P}^2 = Q\bar{P} + QR$, which gives $\bar{P} = \frac{1}{2}(Q + \sqrt{Q^2 + 4QR})$ and $K^* = \bar{P}/(\bar{P}+R) = 1/(1+R/\bar{P})$. Differentiating the Riccati identity gives $\bar{P}' = Q/(2\bar{P} - Q) > 0$, and $\frac{d}{dR}\frac{R}{\bar{P}} = \frac{\bar{P} - R\bar{P}'}{\bar{P}^2} = \frac{2\bar{P}^2 - Q\bar{P} - QR}{\bar{P}^2(2\bar{P}-Q)} = \frac{\bar{P}^2}{\bar{P}^2(2\bar{P}-Q)} > 0$, using $\bar{P}^2 = Q\bar{P} + QR$ twice. Hence K^* is strictly decreasing. The limits follow from $\bar{P} \to Q$ as $R \to 0$ and $\bar{P} \sim \sqrt{QR}$ as $R \to \infty$. For a constant-gain filter, $e_t = \hat{\mu}_t - x_t$ obeys $e_t = (1-K)(e_{t-1} - w_t) + Kv_t$. Independence of the noises gives $S = (1-K)^2(S+Q) + K^2R$, and

Table 25: Fitted update rules complementing Table 2. Each parameter is selected on half the data by error to the oracle-parse Kalman mean and evaluated on the other half. Static discounting applies one variance discount per time step without adding process noise. Dynamic discounting also includes Q and recovers Kalman at $\gamma = 1$. The fixed $\gamma = 0.88$ row is a transferred-parameter sensitivity check. Oracle-input columns isolate the recurrence from extraction; the final column measures public-series error on held-out windows. The static rule's lower series MAE is consistent with dynamics omitted by the random-walk reference.

Update rule (held-out half)	Parameter telemetry / CO_2	Telemetry MAE LLM input	Telemetry MAE oracle input	CO_2 8B deviation	CO_2 8B MAE
Zhang static (fitted)	γ^*=0.39 / 0.66	0.091	0.069	0.090	0.597
Zhang static (frozen γ=0.88)	γ=0.88	0.847	0.839	0.317	0.825
Zhang dynamic (fitted)	γ^*=1.00 / 1.00	0.028	0.000	0.038	0.617
fitted gain	K=0.33 / 0.56	0.736	0.771	0.960	1.128
two-gain	K_A, K_B	0.120	0.082	0.592	0.837
oracle parse → declared Kalman (reference)	γ=1	n/a	0.000	0.000	0.627

Table 26: Sensitivity to the assumed sensor variance on 60 telemetry trajectories. The true R_B is 9. The table separately reports underestimation and overestimation, checkpoint MAE, and terminal alarm agreement. Threefold underestimation gives 0.265 and threefold overestimation gives 0.114, both below end-to-end MAE 0.471.

Assumed $\hat{R}_B$	MAE	Δ vs. exact	alarm agr.
0.33× (3.0)	0.265	+0.248	0.967
0.50× (4.5)	0.151	+0.134	1.000
1.00× (9.0, exact)	**0.017**	+0.000	1.000
2.00× (18.0)	0.088	+0.071	0.983
3.00× (27.0)	0.114	+0.097	0.983
9.00× (81.0)	0.151	+0.134	0.983

Table 27: Window-bootstrap 95% confidence intervals for public-series evaluation. Deviation measures error to the oracle-parse Kalman mean, whereas series MAE measures error to the public observations. Both Qwen checkpoints meet the registered checks. These intervals assess the measured tracking errors, not whether the random-walk model is a correct description of the series.

	Qwen3-8B (N=76)		Qwen3-4B (N=30)	
Arm	dev.	public MAE	dev.	public MAE
end-to-end LLM	0.398 [0.35,0.45]	0.723 [0.65,0.80]	0.451 [0.37,0.55]	0.699 [0.59,0.82]
regex + Kalman	0.170 [0.13,0.21]	0.689 [0.61,0.77]	0.118 [0.08,0.16]	0.659 [0.56,0.76]
PDDA	**0.032 [0.02,0.05]**	**0.605 [0.54,0.67]**	**0.045 [0.01,0.09]**	**0.633 [0.53,0.74]**
last reading†	1.840 [1.63,2.05]	1.908 [1.69,2.14]	1.787 [1.47,2.14]	1.840 [1.51,2.20]
running mean†	0.748 [0.67,0.83]	1.135 [1.02,1.25]	0.774 [0.67,0.88]	1.150 [0.96,1.34]
Kalman (reference)†	0.000 [0.00,0.00]	0.609 [0.55,0.67]	0.000 [0.00,0.00]	0.609 [0.52,0.70]

hence the stated formula for $S(K)$. The Riccati identity also gives $S(K^*) = K^*R = \bar{P} - Q$ and $Q = (\bar{P} + R)(K^*)^2$. Substitution yields the exact factorization

$$S(K) - S(K^*) = \frac{(\bar{P} + R)(K - K^*)^2}{K(2 - K)}.$$

For $K \in (0, 1]$, the denominator is positive. The excess is zero precisely at $K = K^*$ and strictly positive otherwise, proving the unique minimum. The numerical values in the text follow by substitution. □

Proposition 2 (An isolated reading error and a persistent gain mismatch)**.** *For a steady-state filter with gain $K \in (0, 1]$ on the task of §2, replacing a single reading z_s by $z_s + \delta$, while keeping all other readings and updates unchanged, shifts the filtered mean at time $t \geq s$ by $K\delta(1 - K)^{t-s}$. At $K = 0.45$, its half-life is less than 1.2 subsequent updates. In contrast, using a constant gain $K \neq K^*(R)$ gives a steady-state excess MSE of $S(K) - S(K^*) > 0$.*

Proof of Proposition 2. For the first claim, compare two filters with identical gains and inputs except for $z_s \mapsto z_s + \delta$. Their means differ by $K\delta$ immediately after that update. Each subsequent update multiplies the difference by $1 - K$, giving $K\delta(1 - K)^{t-s}$. At $K = 0.45$, the half-life is $\ln 2/|\ln 0.55| \approx 1.16$ updates. If gains vary but remain identical between the two compared runs, the difference is $K_s\delta \prod_{u=s+1}^{t}(1 - K_u)$. It decays geometrically over valid observation updates when those gains have a positive lower bound. For the single-observation-per-step scalar system with $Q > 0$ and bounded observation variance, $P_u^- \geq Q$ before an observed time step and hence $K_u \geq Q/(Q + R_{\max})$. During an observation gap there is no contraction update. Removing a reading or assigning it to a different sensor can change the gains themselves, so such errors are not covered by this comparison. For the second claim, Proposition 1 gives the unique minimizer of stationary error. Thus $K \neq K^*(R)$ implies $S(K) - S(K^*(R)) > 0$ as long as the mismatched stationary rule is used. □

M ARCHIVED SHORT-BUDGET THINKING CONTROL

The original control uses Qwen3-8B's thinking template with 40 trajectories at each of $R \in \{1, 16\}$ and a $1{,}024$-token budget. Endpoint seeds depend on R, so the endpoints are not paired. The parser strips text through the closing `</think>` tag when present; it does not enforce closure before attempting parsing. Only $2/600$ beliefs parse at $R = 1$ and none at $R = 16$, below the registered requirement for assessing adaptation. This short-budget, trajectory-level control is a different protocol from the fully parsed fixed-history comparison in App. B.3.