DARC: DECISION-AWARE READOUT CALIBRATION FOR GENERATIVE WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

A planner compares candidate actions through a statistic of imagined futures. Numerical solver accuracy measures approximation to the learned law; accuracy of that statistic against a simulator reference is a separate question. Matched-noise solver ladders and reset-and-replay measurements show that finite-step contraction can partly cancel learned-model over-dispersion; improving the solve can therefore worsen the planner's readout. We propose **DARC**, a post-hoc method that diagnoses a frozen generative world model against a specified simulator-replay law and fits a mean-preserving value-sample dispersion scalar directly in the planner's readout space. On 135 trained latent models, readout fitting removes 25.0% of readout error (hierarchical 95% interval [15.8, 35.0]) and harms 1 of 1620 evaluated cells by more than 0.5%; the same transport fitted to latent variance changes mean error by -16.5% and harms 330 cells. Pixel models show positive capped-mean gains, although their ordering is sensitive to the aggregation metric. A simulator-reference candidate-choice audit does not establish a ranking or regret gain, and the closed-loop tests do not establish improved control. The online correction uses no additional world-model evaluations.

1 INTRODUCTION

Generative world models support planning by predicting future observations under candidate actions (Ha & Schmidhuber, 2018; Hafner et al., 2019; 2020; 2023; Janner et al., 2022). At an anchor state, a planner constructs a set of actions, samples several futures for each action, maps every future through a value head, and aggregates the resulting values with a task readout such as value, tail risk, failure probability, or interval width. The planner therefore consumes a distributional statistic and the ordering of candidate scores. Frame similarity and solver accuracy remain useful diagnostics, while the quantity that enters action selection is the planner readout. This distinction defines *readout fidelity*: agreement with a reference law after the complete transformation used by the planner.

Increasing the integration budget improves approximation to the learned generative law $Q_{\theta,\infty}$. Environment agreement is measured against a different conditional law, P_{ref}, which we construct by reset-and-replay under a specified actuator-noise process. The finite-step law $Q_{\theta,S}$ can lie between those two distributions. When the learned model is over-dispersed, finite-step contraction can cancel part of the model error; solver convergence then removes the cancellation and exposes a larger simulator-referenced discrepancy. In the measured locomotion and Jaco settings, longer integration yields sharper samples while the planner-readout error reaches 172% and 131% of the corresponding one-step baselines. Numerical, environment, and readout fidelity consequently describe different comparisons and can move in different directions.

These observations raise three linked questions. ① **Error provenance and direction.** How can a frozen world model separate learned-model discrepancy from finite-inference discrepancy, and which variation changes comparisons among candidate actions? ② **Readout-specific calibration.** Given a measured discrepancy, how should its direction and magnitude be selected for the value or risk statistic used by the planner, with independent data controlling selection and acceptance? ③ **Decision consequence.** Which conditions connect a reduction in readout error to preservation of a reference action, and do the measured estimation gains extend to candidate ranking or closed-loop control? The first question identifies the source and structure of the error, the second defines a correctable intervention, and the third fixes the scope of the empirical claim.

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.

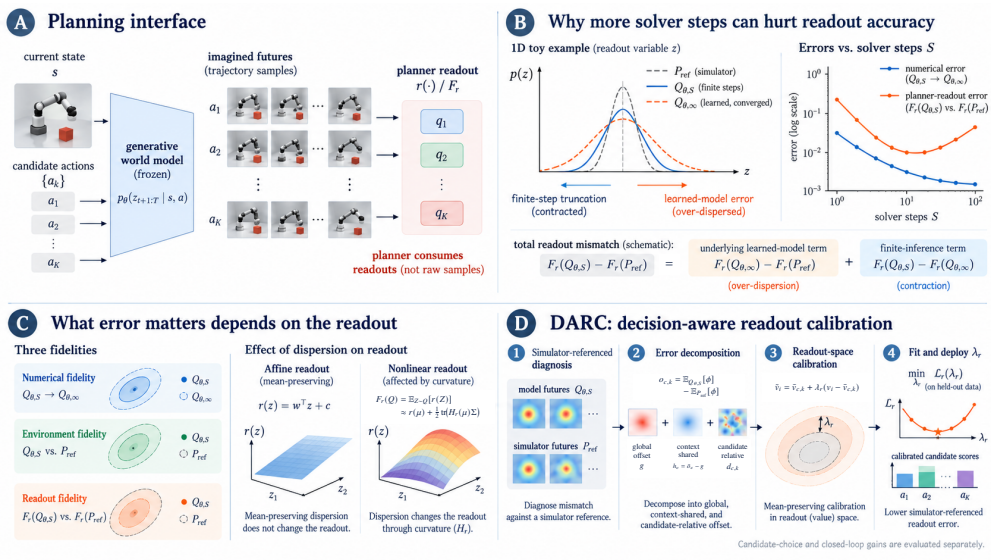


Figure 1: **DARC overview: planning readouts, error sources, and calibration.** **(A)** A frozen world model samples futures under candidate actions; the planner reduces each set of futures to an action score. **(B)** Finite-step contraction can partly offset learned-model over-dispersion, so numerical error may fall while simulator-referenced readout error rises. **(C)** Numerical, environment, and readout fidelity compare distinct quantities. Mean-preserving value-sample rescaling leaves expected affine readouts unchanged, while nonlinear readouts can respond through curvature. **(D)** Matched simulator replay identifies candidate-indexed offsets; DARC separates shared and relative components, fits a scalar in readout space on disjoint contexts, and applies it under the frozen model without extra world-model evaluations. Curves in (B) and the loss in (D) are schematic; candidate-choice and closed-loop results are evaluated separately.

DARC answers this sequence with a frozen-model pipeline. Shared initial noise across a solver ladder isolates finite-inference error, and matched simulator replays define the environment reference. An empirical orthogonal decomposition separates global, context-shared, and candidate-relative offsets. DARC then applies a mean-preserving transport to value samples and selects a scalar $\lambda_{t,S,r}$ for each task, solver budget, and readout by minimizing the simulator-referenced readout error on D_{fit}. A disjoint D_{gate} screens the fitted scalar, and D_{test} is used once for reporting. An accepted scalar acts after the frozen value head and preserves the original number of world-model evaluations.

The evaluation treats estimation, candidate selection, and control as separate endpoints. Across 135 trained latent models, readout fitting removes 25.0% of measured readout error with a hierarchical 95% interval of [15.8, 35.0] and harms 1 of 1620 cells by more than 0.5%. Fitting the same transport to latent variance changes mean error by -16.5% and harms 330 cells. Pixel-model comparisons test the same target-alignment question, although their family ordering changes with the aggregation metric. The candidate audit gives overlapping top-1, pairwise, Kendall, and regret results for DARC and the plain sampler. Closed-loop tests likewise yield no established return or success gain. The supported empirical claim is improved estimation of the tested planner readouts under the stated replay protocol.

Contributions. **(1) Three-law formulation and matched error attribution.** We distinguish finite-step, converged-model, and simulator-reference laws under explicit replay. Shared initial noise separates solver and model discrepancies; an orthogonal decomposition locates global, context-shared, and action-relative variation without assuming which component affects decisions. **(2) Readout-specific calibration with independent selection.** DARC introduces a mean-preserving value-sample transport and fits its scalar to the deployed readout on disjoint fit, gate, and test contexts. Generator, value head, and sampling budget stay fixed. **(3) Theory and endpoint-complete evaluation.** We establish affine inertness, characterize nonlinear curvature response, opposing latent and readout

optima, and a sufficient candidate margin. Experiments report estimation gains, harmed cells, aggregation sensitivity, predicted nulls, ranking, and closed-loop control.

2 Related work

World models and decision-aware objectives. Latent imagination and model predictive control use learned dynamics for action selection (Hafner et al., 2021; Hansen et al., 2022; 2024); visual representations and embodied settings broaden their scope (Zhang et al., 2019; Seo et al., 2023; Wu et al., 2023; Sun et al., 2024; Agarwal et al., 2024). Probabilistic ensembles and short model rollouts address model uncertainty and accumulated bias (Chua et al., 2018; Janner et al., 2019). Value-aware objectives and policy-relevant model learning move the training target toward decisions (Farahmand et al., 2017; Grimm et al., 2020; 2021; Schrittwieser et al., 2020; Ma et al., 2023; Voelcker et al., 2025). DARC instead holds the trained generator and value head fixed and fits the statistic used by its existing planner against a specified simulator replay law.

Generative solvers and inference budgets. Diffusion, latent diffusion and flow modeling define distinct learned laws (Ho et al., 2020; Song et al., 2021b; Rombach et al., 2022; Lipman et al., 2023). Deterministic samplers, specialized ODE solvers, distilled samplers and consistency models change the number of evaluations needed to approach those laws (Song et al., 2021a; Karras et al., 2022; Lu et al., 2022; Salimans & Ho, 2022; Song et al., 2023). Diffusion policies and adaptive allocation change how inference work is spent (Janner et al., 2022; Chi et al., 2023; Le et al., 2026; Li et al., 2026). DARC tests whether solver convergence also improves the simulator-referenced planner readout, then calibrates that readout at a fixed world-model budget.

Calibration and fidelity endpoints. Probability, regression and model-based uncertainty calibration target different forecast properties (Guo et al., 2017; Kuleshov et al., 2018; Malik et al., 2019); ensembles and distribution-shift tests expose uncertainty-estimation limits (Lakshminarayanan et al., 2017; Ovadia et al., 2019). Proper scoring rules, two-sample tests and tail-risk criteria clarify why the evaluation target must be named (Gneiting & Raftery, 2007; Gretton et al., 2012; Rockafellar & Uryasev, 2000). Perceptual quality and distortion can disagree (Blau & Michaeli, 2018). We report image fidelity, readout error, candidate ranking and closed-loop outcomes separately.

3 Proposed Framework: DARC

3.1 Problem Formulation and Framework Overview

Let an anchor context be $c = (s_c, \mathcal{A}_c)$, where $\mathcal{A}_c = \{a_{c,1}, \dots, a_{c,K}\}$ is the candidate set. For candidate k, the frozen S-step world model defines a trajectory law $Q^{c,k}_{\theta,S}$, and reset-and-replay under noise process ν defines the conditional reference $P^{c,k;\nu}_{\mathrm{ref}}$. A frozen value head V maps trajectories to value-sample laws $\mu^{c,k}_{\theta,S} = V_{\#}Q^{c,k}_{\theta,S}$ and $\mu^{c,k}_{\mathrm{ref}} = V_{\#}P^{c,k;\nu}_{\mathrm{ref}}$. Readout F_r then gives the model and reference candidate scores

$$q_{c,k,r}(\lambda) = F_r\Big((T_\lambda)_{\#}\mu^{c,k}_{\theta,S}\Big), \qquad q^*_{c,k,r} = F_r\Big(\mu^{c,k}_{\mathrm{ref}}\Big), \qquad \widehat{a}_c(\lambda) = \arg\max_k q_{c,k,r}(\lambda). \quad (1)$$

The replay protocol and ν determine the scope of P_{ref}. Numerical fidelity compares $Q^{c,k}_{\theta,S}$ with $Q^{c,k}_{\theta,\infty}$, environment fidelity compares either model law with $P^{c,k;\nu}_{\mathrm{ref}}$, and readout fidelity compares their images under V and F_r. Each metric therefore refers to a different pair of objects.

For task t, solver budget S, and readout r, DARC selects a positive scalar from a fixed grid Λ by minimizing the empirical reference loss

$$\widehat{L}_{t,S,r}(\lambda; D) = \frac{1}{|D|}\sum_{(c,k)\in D} \ell_r\big(q_{c,k,r}(\lambda), q^*_{c,k,r}\big), \qquad \widehat{\lambda}_{t,S,r} = \arg\min_{\lambda\in\Lambda} \widehat{L}_{t,S,r}(\lambda; D_{\mathrm{fit}}). \quad (2)$$

Readout loss is the primary optimization endpoint. Candidate agreement, reference regret, and closed-loop return are evaluated separately because a reduction in average score error does not supply the uniform candidate-wise bound required for action preservation.

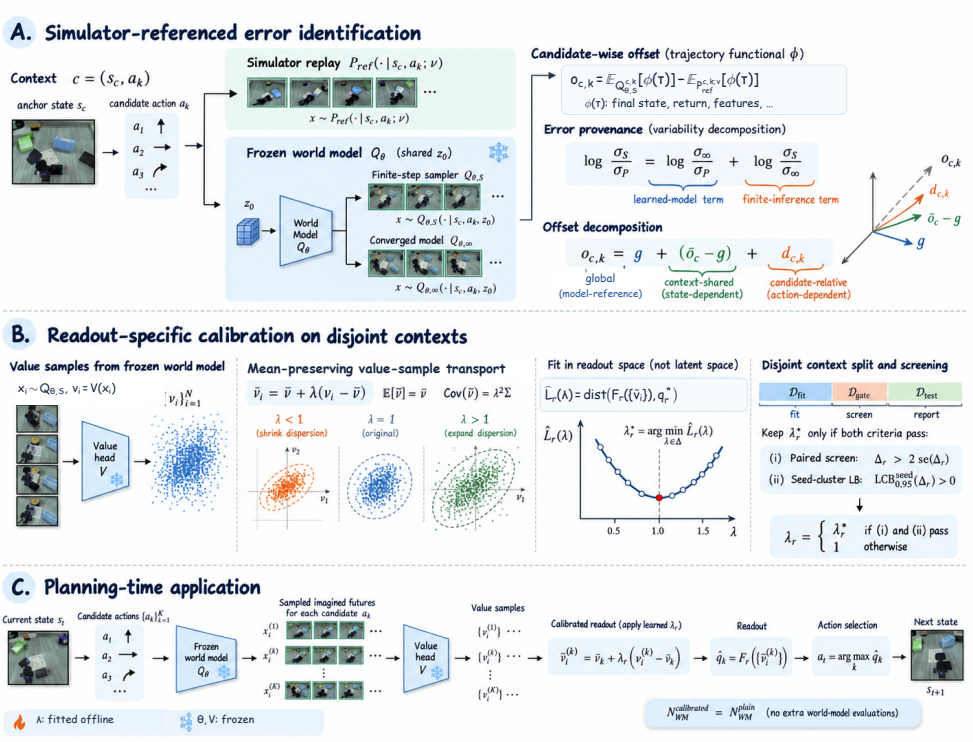


Figure 2: **DARC from diagnosis to planning. (A)** Matched simulator replays define P_{ref}; shared noise yields finite-step and converged model laws. Trajectory offsets are decomposed into global, context-shared, and candidate-relative terms. **(B)** A frozen value head supplies v_i; the mean-preserving map $\tilde{v}_i = \bar{v} + \lambda(v_i - \bar{v})$ is fitted in the deployed readout on disjoint fit, gate, and test contexts. **(C)** The accepted scalar acts before candidate scoring without additional world-model evaluations.

Five stages organize the method: **(1) formulation** specifies three laws, value samples, readout, and data roles; **(2) direction identification** attributes error with matched noise and replays; **(3) adaptive calibration** fits a mean-preserving intervention to the readout; **(4) optimization and application** screens and deploys its scalar with θ and V frozen; and **(5) analysis** determines readout response and candidate margins. Figure 2 illustrates the sequence; Appendix Table 2 defines symbols.

3.2 Simulator-Referenced Direction Identification

Reference construction and error provenance. For anchor context c and candidate k, $P_{\mathrm{ref}}^{c,k;\nu}$ is the conditional law induced by the stated per-step actuator-noise process ν. Exact state restoration followed by R independent replays estimates this law. The converged model law $Q_{\theta,\infty}^{c,k}$ is the pushforward of $z_0 \sim \mathcal{N}(0, I)$ through the solution of the learned probability-flow ODE, and $Q_{\theta,S}^{c,k}$ uses an S-step discretization of that same flow. Sharing z_0 across solver settings removes Monte Carlo variation from their difference. For trajectory functional ϕ, we estimate the candidate offset $o_{c,k} = \mathbb{E}_{Q_{\theta,S}^{c,k}}[\phi] - \mathbb{E}_{P_{\mathrm{ref}}^{c,k;\nu}}[\phi]$ on one replay half and score it on the other. For any positive dispersion statistic σ, the total discrepancy has the exact identity

$$\underbrace{\log \frac{\sigma_{c,k,S}}{\sigma_{c,k,\mathrm{ref}}}}_{\delta^{\mathrm{total}}_{c,k,S}} = \underbrace{\log \frac{\sigma_{c,k,\infty}}{\sigma_{c,k,\mathrm{ref}}}}_{\delta^{\mathrm{model}}_{c,k}} + \underbrace{\log \frac{\sigma_{c,k,S}}{\sigma_{c,k,\infty}}}_{\delta^{\mathrm{solver}}_{c,k,S}}. \tag{3}$$

Opposite signs indicate cancellation and equal signs indicate reinforcement. Equation (3) attributes the discrepancy; the readout objective in Eq. (2) determines the calibration direction.

Admissibility diagnostics. We measure conditional spread against replay noise and the top-pair ratio $\gamma = (Q_{(1)} - Q_{(2)})/\operatorname{se}(Q)$, requiring $\gamma > 2$ to interpret an ordering. Per-step actuator noise yields resolved spread and rank across three protocols; holding one action throughout makes spread nearly degenerate, while random actions after one candidate action leave rank unresolved. Attribution proceeds only where the reference is informative.

Candidate-wise structure. Let $\bar{o}_c = K^{-1}\sum_k o_{c,k}$, $g = (CK)^{-1}\sum_{c,k} o_{c,k}$, $h_c = \bar{o}_c - g$, and $d_{c,k} = o_{c,k} - \bar{o}_c$. Then $o_{c,k} = g + h_c + d_{c,k}$ and the three components are orthogonal under the empirical inner product over contexts and candidates:

$$\sum_{c,k} \|o_{c,k}\|^2 = \underbrace{CK\|g\|^2}_{\text{global}} + \underbrace{K\sum_{c} \|h_c\|^2}_{\text{context shared}} + \underbrace{\sum_{c,k} \|d_{c,k}\|^2}_{\text{candidate relative}}. \tag{4}$$

This squared-offset identity locates where an error varies. Energy distance between distributions is reported as an unsplit scalar because it lacks the additive identity in Eq. (4). Candidate-relative offsets enter contrasts directly at the chosen representation, while shared offsets affect score contrasts when downstream sensitivity varies across candidates.

Proposition 1 (Feature-space shift and downstream sensitivity)**.** *Let X_i and X_j be two candidate feature laws, let r be the per-future map that precedes aggregation, and define $F_i(b) = \mathbb{E}[r(X_i + b)]$. For affine r, the difference $F_i(b) - F_j(b)$ is invariant to a shared b. A nonlinear r can change that difference: with $r(z) = z^2$ and $X_\ell \sim \mathcal{N}(m_\ell, s_\ell^2)$ for $\ell \in \{i, j\}$, the candidates exchange order at*

$$b^* = \frac{(m_i^2 - m_j^2) + (s_i^2 - s_j^2)}{2(m_j - m_i)}, \qquad m_i \neq m_j. \tag{5}$$

The proposition concerns an upstream feature shift followed by a per-future nonlinear map. DARC's deployed transport acts on value samples after V, so every offset component is evaluated through the actual planner readout F_r. Direction identification therefore returns an attributed, candidate-indexed discrepancy; the next stage selects a correction from measured readout loss.

3.3 Readout-Specific Adaptive Calibration

For candidate value samples $v_1, \ldots, v_n$ with sample mean $\bar{v}$, DARC applies

$$T_\lambda(v_i; \bar{v}) = \bar{v} + \lambda(v_i - \bar{v}), \qquad \lambda \in \Lambda \subset \mathbb{R}_+. \tag{6}$$

Centering is performed separately for every context and candidate. The transform preserves $\bar{v}$ exactly and scales the sample covariance by λ^2; $\lambda < 1$ contracts dispersion, $\lambda > 1$ expands it, and $\lambda = 1$ recovers the uncalibrated samples. This family follows the identified spread discrepancy while leaving the conditional mean unchanged. It makes no claim of repairing the complete trajectory or image distribution.

The response depends on the readout. The sample mean is an exact negative control, while interval width, dispersion, tail value, and bad-outcome probability can change under Eq. (6). DARC therefore fits $\lambda_{t,S,r}$ separately for every deployed readout using Eq. (2); a composite planner objective receives one scalar fitted directly to that composite. Fitting the same family to latent variance supplies the target-mismatch baseline. The scalar is shared across held-out contexts, seeds, and trained configurations within a task, solver step, and readout group, which tests whether one correction can be amortized over the contexts in that group.

3.4 Model Optimization and Frozen-Model Application

The optimization variable is the calibration scalar; world-model parameters θ and value-head parameters remain fixed. D_{fit} selects $\widehat{\lambda}_{t,S,r}$ through Eq. (2). On the disjoint gate split, the paired improvement is

$$\Delta_{t,S,r} = \widehat{L}_{t,S,r}(1; D_{\text{gate}}) - \widehat{L}_{t,S,r}(\widehat{\lambda}_{t,S,r}; D_{\text{gate}}). \tag{7}$$

The scalar is retained when the within-model paired screen $\Delta_{t,S,r} > 2\operatorname{se}(\Delta_{t,S,r})$ passes. A supplemental dependence audit vetoes a task, step, and readout group unless the one-sided 95% seed-cluster

bootstrap lower bound for its held-out mean gain is positive. Resampling keeps each seed's nine configurations and validation contexts together. With three seed clusters per task, this second screen serves as an empirical veto with limited uncertainty resolution. Failure of either screen sets $\lambda_{t,S,r} = 1$. D_{test} is evaluated after every selection is frozen.

At planning time, candidate k is scored as $\widehat{q}_{c,k,r} = F_r(\{\bar{v}_{c,k} + \lambda_{t,S,r}(v_{c,k}^{(i)} - \bar{v}_{c,k})\}_{i=1}^{n})$, and the controller executes $\arg\max_k \widehat{q}_{c,k,r}$. Computing the centered transform costs $O(n)$ scalar arithmetic for n value samples and preserves the original number of world-model calls. The calibration table is indexed by task, sampler budget, and readout, so deployment does not open the fit or gate data again.

3.5 Theoretical Analysis

The propositions distinguish responsive readouts, latent and readout optima, and a sufficient condition for preserving the reference action. Appendix B gives proofs; experiments separately test curvature, ranking, and closed-loop outcomes.

Proposition 2 (Transport identities and affine inertness)**.** *Let Q have mean m, covariance Σ and finite second moment. For every $\lambda \in \mathbb{R}$:* (i) $\mathbb{E}[T_\lambda \# Q] = m$ *and* $\mathrm{Cov}[T_\lambda \# Q] = \lambda^2 \Sigma$*;* (ii) *if the readout is $F_r(Q) = \mathbb{E}_Q[r(Z)]$ with affine $r(z) = w^\top z + c$, then $F_r(T_\lambda \# Q) = w^\top m + c$, independent of λ.*

Proposition 3 (Curvature transport)**.** *Let r be twice differentiable and $\delta = X - m$. Then $G_r(\lambda) = \mathbb{E}[r(m + \lambda\delta)] = r(m) + \frac{\lambda^2}{2}\,\mathrm{tr}[H_r(m)\Sigma] + o(\lambda^2)$, with equality when r is quadratic.*

Proposition 4 (Distinct latent and decision optima)**.** *Let $P_{\text{ref}} = \mathcal{N}(0,1)$, $Q = \mathcal{N}(b, \sigma^2)$ and $r(z) = z^2$. Then $\lambda_{\text{lat}} = \arg\min_\lambda \mathrm{KL}(T_\lambda \# Q \,\|\, P_{\text{ref}}) = 1/\sigma$, independent of b, while $\lambda_{\text{dec}} = \arg\min_\lambda |F_r(T_\lambda \# Q) - F_r(P_{\text{ref}})| = \sqrt{1 - b^2}/\sigma$ for $|b| \le 1$. For $b \neq 0$ they differ, and each is strictly worse under the other's objective.*

Proposition 5 (Candidate selection under bounded score error)**.** *For reference candidate scores q_k^* and estimates $\tilde{q}_k$, let $k^* = \arg\max_k q_k^*$, $\hat{k} = \arg\max_k \tilde{q}_k$, and $\epsilon_k = |\tilde{q}_k - q_k^*|$. If the best-to-second-best reference margin γ exceeds $2\max_k \epsilon_k$, then $\hat{k} = k^*$. Without a margin assumption, the reference regret satisfies $q_{k^*}^* - q_{\hat{k}}^* \le \epsilon_{k^*} + \epsilon_{\hat{k}} \le 2\max_k \epsilon_k$.*

Affine readouts are exact negative controls. For smooth nonlinear readouts, the leading response couples covariance and curvature through $\mathrm{tr}[H_r\Sigma]$. At $b = 0.8$ and $\sigma = 0.7$, Proposition 4 gives latent and decision optima of 1.43 and 0.86, on opposite sides of one. Proposition 5 requires uniform candidate-wise accuracy; an average score-error reduction cannot alone preserve action choice.

Implication for compute allocation. A fixed sampler budget depends on the marginal return of one more step; the current error level alone is insufficient. Appendix G derives and measures this distinction. DARC calibrates at a fixed budget and leaves the sampling schedule unchanged.

4 Experiments

4.1 Evaluation Protocol

World models and environments. We evaluate 135 trained latent world models (5 tasks $\times$ 3 seeds $\times$ 9 configurations spanning training-set size, network width, and training duration), a patch-token video model, and two pixel-space video models trained from scratch. A 64M-parameter flow-matching U-Net covers nine dm_control environments: walker, cheetah, reacher, cartpole, pendulum, acrobot, ball-in-cup, finger and hopper. A second model of the same architecture covers four visually distinct Jaco manipulation scenes (Tassa et al., 2018; Todorov et al., 2012). Per-step actuator noise defines the simulator conditional law P_{ref} for matched replays.

Readouts and statistics. We evaluate the worst value quartile (CVaR_{25}), bad-outcome probability, central-interval width and dispersion over sampled futures. These nonlinear readouts can respond to a mean-preserving spread change; the plain mean is retained as a predicted null control. Cells share tasks, seeds and checkpoints, so confidence intervals for the latent-model sweep resample task, seed, and trained configuration while keeping all step/readout cells of a model together.

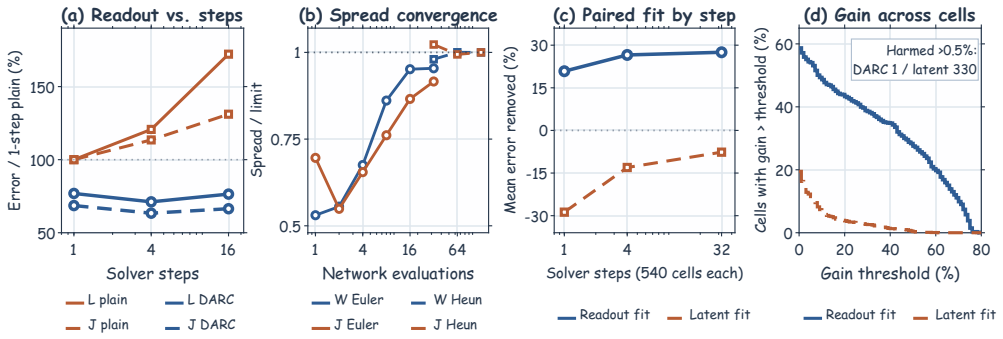


Figure 3: **Ordered-budget trends and paired-cell effects. (a)** Video-model readout error relative to each family's one-step plain arm (L: six locomotion scenes; J: four Jaco scenes; lower is better). **(b)** Spread relative to the converged learned law, not the simulator law. **(c)** Mean error removed at three budgets (540 paired latent cells each). **(d)** Share of 1620 cells exceeding each gain threshold; inset counts cells harmed by more than 0.5%. Lines join measured budgets or empirical threshold values only.

A cell is *harmed* if corrected readout error rises by more than 0.5%. Throughout, S is solver steps and n is Monte Carlo futures per candidate.

4.2 RQ1: Solver Convergence and Reference Readout Fidelity

Observation 1: solver convergence can expose learned-model readout error. At the measured solver budgets, Figure 3(a) shows plain video-model readout error rising from 100 to 172.2% on locomotion and 131.1% on Jaco as S grows from 1 to 16; DARC stays below 77% and 69%, respectively, at the measured budgets. Panel (b) checks convergence to each learned law. Holding z_0 fixed and applying Eq. (3) then attributes the simulator-referenced discrepancy (Appendix E). The converged model is over-dispersed in 43 of 48 cells, median ratio 3.0 to 3.9. The learned-model term accounts for 63% (95% CI [46, 82]) of the measured log-dispersion discrepancy on walker and 78% ([71, 87]) on the arm, under the stated replay protocol. In 41 of the 48 cells the finite-inference term carries the opposite sign: forward-Euler truncation contracts the sampled spread in the direction the model's over-dispersion needs correcting, so a coarse solver masks part of the learned model's error because the two terms have opposite signs. This cancellation produces the mechanism illustrated in Figure 1(B).

4.3 RQ2: Readout-Aligned Calibration and Estimation

Observation 2: the optimization target determines both gain and harm. Figure 3(c) resolves the 1620-cell comparison by sampler budget: readout fitting removes 20.8%, 26.6%, and 27.5% mean error at 1, 4, and 32 steps, whereas latent fitting changes error by -28.8%, -13.0%, and -7.7%. Pooled readout gain is +25.0% (hierarchical 95% interval [+15.8, +35.0]) versus -16.5% ([-40.1, -2.5]) for latent fitting. Panel (d) shows the positive-gain survival curves and the 1 versus 330 harmed cells. Direction can reverse: in 20 of the 77 cells the gate accepts, λ_{spread} and λ_r lie on opposite sides of one. On the manipulation family the latent criterion harms 1129 of 3456 cells against 477 for the readout. The paired-gate ablation in Table 1 isolates the screening rule, and Figure 4 shows the candidate-relative component in four pixel-space scenes; the full spatial decomposition in Appendix C also shows the shared scene-wide component.

4.4 Calibratable Error Structure

Across the seven cells with a nonzero per-context optimum, one shared constant recovers 94.7% of the available gain, compared with 30.7% for a per-context encoder estimate whose finite-sample variance offsets its flexibility. For conditional mean bias, a global constant, ridge, and MLP each remove at most 2.6%, with explained variance at most 3.7%. Under the tested classes, dispersion is therefore

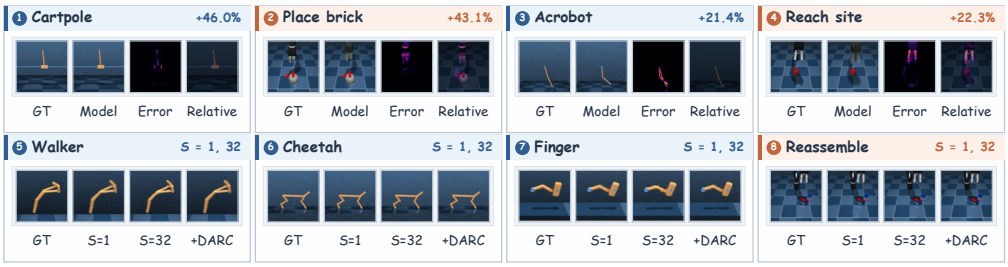


Figure 4: **Matched visual cases.** Eight equal-size cards each contain four crops. Upper row: simulator (GT), prediction (Model), pixel error (Error), and candidate-relative overlay (Relative); percentages are measured readout-error gains. Lower row: simulator, one-step ($S = 1$), 32-step ($S = 32$), and the one-step rollout scored by DARC (+DARC). DARC calibrates the readout after image generation and does not alter the rollout. Blue and orange headers denote locomotion and manipulation; full sequences appear in Appendix C.

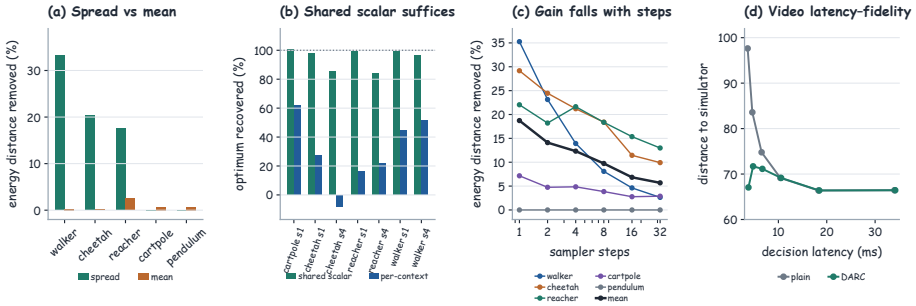


Figure 5: **What calibration can amortize.** **(a)** Shared spread correction removes 33.2%, 20.4%, and 17.5% error on three tasks; mean correction removes at most 2.6%. **(b)** One constant recovers 84–101% of the context optimum versus −8–62% for context estimates. **(c)** Mean gain falls from 18.7% at one step to 5.7% at 32. **(d)** One calibrated video-model step matches 32 plain steps at 8.9× lower latency.

more consistently amortizable than conditional mean bias. The candidate-relative component carries 51.4% of offset energy on the latent sweep and 65% and 92% on the video models.

Removing it improves all 20 measured cells. A shared shift can reorder candidates through nonlinear readouts; its measured decision-error effect is small.

Figure 5 tests the shared scalar and its declining marginal value with more solver steps; Appendix L gives context-level and latency measurements.

4.5 Ablations, Screening, and Predicted Nulls

Table 1 isolates target, gate, and family. Latent fitting reverses mean gain and harms 330 versus 1 cell; gate removal raises gain but harms 24. Neither an affine nor a higher-capacity monotone map consistently exceeds the scalar.

Cluster-aware screening and robust aggregation. A seed-cluster veto retains 53/60 groups, shifts mean gain from 24.97% to 24.93%, and leaves one harmed cell. Readout fitting exceeds latent fitting in five task and 15 task–seed aggregates; 20/77 accepted cells have opposing fitted directions (Figure 17). Pixel-family ordering depends on aggregation: robust medians slightly favor latent fitting on locomotion, while capped means favor readout fitting in both families.

Value-head blur control. Across the one-to-32-step blur range, value-head bias shifts only 0.0008 normalized units; variance grows toward blurrier samples and thus works against DARC (Appendix F.1).

Table 1: **Ablations and matched comparisons.** (a) One component changes at a time. (b) Both fits use the same cells. Gain is readout error removed; harm is a paired increase greater than 0.5%. A dash denotes an unreported harmed-cell count.

(a) Component ablations

Component	Variant	Gain	Harmed
Fit target	latent variance	−16.5%	330/1620
	readout	**+25.0%**	**1/1620**
Paired gate	removed	+61.9%	24/156
	kept	+43.3%	**0/156**
Family	affine map	19.6%	—
	monotone map	2.3%	—
	scalar	18.2%	—

(b) Matched family comparison

Evaluation set	Readout	Latent
Latent sweep	+25.0%	−16.5%
Locomotion	+35.0%	+19.7%
Jaco	+35.1%	−10.2%
Harmed cells: readout / latent		
Latent sweep	1/1620	330/1620
Locomotion	510/5184	1336/5184
Jaco	477/3456	1129/3456

Predicted affine null. For an affine readout, Proposition 2(ii) predicts inertness. The mean-readout gain and standard error stay below 1.5×10^{-6} across nine cells and pass no gate; nonlinear shape readouts gain 0.4–49.4% and pass in 17/18 cells.

4.6 RQ3: Candidate Choice and Closed-Loop Evaluation

Observation 3: estimation gains do not establish ranking or control gains.

Candidate choice under an independent simulator reference. Proposition 5 requires a uniform candidate-wise bound, so we test five actions per held-out context against the 24-replay simulator half excluded from fitting. Over 135 models at four steps, plain and DARC top-1 CVaR agreement is 29.41% and 29.43%; pairwise agreement is 56.96% for both, and reference regret is 0.155799 and 0.155804 reward units. Paired task, seed, and configuration bootstrap intervals for DARC minus plain are [-0.52, +0.71] top-1 points and [-0.00010, +0.00014] regret units. The two reference halves agree on the top action in 69.4% of contexts (Appendix Figure 20). PointMaze success is 43.3±6.4% plain versus 41.7±6.4% DARC across 60 episodes per arm; walker and cheetah returns across 100 episodes per arm show no tested advantage (Appendix K). The observed estimation gain therefore supports neither a candidate-selection nor a control-return gain.

5 Limitations

Attribution assumes deterministic flow sampling and identical-state reset-and-replay; its reference depends on the specified actuator noise. The perturbation study covers two tasks and one horizon, and the three-seed gate offers no out-of-distribution guarantee. The 48-cell attribution covers two scenes and one readout; replay-prefix signs persist, but absolute ratios remain unconverged. Energy shares depend on representation, and fitted scalars may not transfer to new tasks or readouts.

6 Conclusion

Planner fidelity depends on the statistic scoring candidate actions. The three-law formulation separates finite-step, converged-model, and simulator-reference laws. Matched solver ladders show that finite-step contraction can cancel learned-model over-dispersion, exposing readout error as the solver converges. Candidate-wise decomposition locates the discrepancy; the value head and readout determine its effect on action scores.

DARC applies a mean-preserving transport after the frozen value head, fits it in readout space, and screens it on disjoint contexts without extra model evaluations. Across 135 latent models, it removes 25.0% of measured error, versus −16.5% for latent fitting. Affine inertness and nonlinear curvature delimit responsive readouts. Ranking, regret, and closed-loop tests show no reliable gain despite better estimation, as the candidate-margin condition permits. Stronger candidate-wise bounds and replay robustness are needed before claiming better planning.

AI USE STATEMENT

Large language models assisted with experimental code, plot generation, LaTeX editing, numerical audits, theorem checking, and prose revision. The authors executed the analyses, compared newly generated outputs with stored experiment records, reviewed the derivations and interpretations, and take responsibility for the research design and submission. No numerical result in the paper is sourced from generated text; reported values come from executed code and stored simulator or model outputs.

ETHICS STATEMENT

This work studies simulated control environments and involves no human subjects, no personal data and no deployment on physical hardware. The main foreseeable risk is overclaiming: a calibration that improves a planner's statistic on a calibration distribution could be mistaken for a safety guarantee. We therefore state explicitly that the gate of §3.3 is a held-out screening rule and not a distributional guarantee, that the closed-loop measurements of Appendix K are reported as a null result and not as a claim that the calibration improves control, and that nothing here supports a claim about physical robot performance.

REPRODUCIBILITY STATEMENT

The supplementary source includes analysis scripts, figure-generation code, and the plotted numerical summaries. Appendix A states the solver ladders, replay counts, context partitions, and readout definitions; Appendix B gives complete proofs. Five latent tasks use three training seeds and nine trained configurations each. The hierarchical-bootstrap scripts state their resampling units, and the candidate-choice audit reports the number of held-out contexts, candidates, and independent simulator replays. The Overleaf source package excludes raw video tensors, model checkpoints, and the large per-candidate record store; reproducing those measurements requires the original training and simulator artifacts. Independent simulator replays use the stated per-step actuator-noise protocol to define P_{ref}.

REFERENCES

Pranav Agarwal, Sheldon Andrews, and Samira Ebrahimi Kahou. Learning to play atari in a world of tokens. In *Proceedings of the 41st International Conference on Machine Learning*, 2024. URL https://proceedings.mlr.press/v235/agarwal24b.html.

Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Robotics: Science and Systems (RSS)*, 2023.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems*, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/3de568f8597b94bda53149c7d7f5958c-Abstract.html.

Amir-massoud Farahmand, André Barreto, and Daniel Nikovski. Value-aware loss function for model-based reinforcement learning. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: 10.1198/016214506000001437.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012. URL https://www.jmlr.org/papers/v13/gretton12a.html.

Christopher Grimm, André Barreto, Satinder Singh, and David Silver. The value equivalence principle for model-based reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Christopher Grimm, André Barreto, Gregory Farquhar, David Silver, and Satinder Singh. Proper value equivalence. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018. URL https://worldmodels.github.io/.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *Proceedings of the 36th International Conference on Machine Learning*, 2019. URL https://proceedings.mlr.press/v97/hafner19a.html.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=S1lOTC4tDS.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=0oabwyZbOu.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Oxh5CstDJU.

Nicklas A. Hansen, Hao Su, and Xiaolong Wang. Temporal difference learning for model predictive control. In *Proceedings of the 39th International Conference on Machine Learning*, 2022. URL https://proceedings.mlr.press/v162/hansen22a.html.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. In *Advances in Neural Information Processing Systems*, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/5faf461eff3099671ad63c6f3f094f7f-Abstract.html.

Michael Janner, Yilun Du, Joshua B. Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, 2022.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *Proceedings of the 35th International Conference on Machine Learning*, 2018. URL https://proceedings.mlr.press/v80/kuleshov18a.html.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/9ef2ed4b7fd2c810847ffa5fa85bce38-Abstract.html.

Thanh-Tung Le, Yunhan Zhao, Menglei Chai, Zhengyang Shen, Zhe Cao, Danhang Tang, Xiaohui Xie, and Deying Kong. DSA: Dynamic step allocation for fast autoregressive video generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Findings Track*, 2026. arXiv:2606.04432.

Andrew Zou Li, Gokul Swamy, Yonatan Bisk, and Andrea Bajcsy. ELASTIC: Efficiently learning to adaptively scale test-time compute for generative control policies. *arXiv preprint arXiv:2606.31132*, 2026.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Yecheng Jason Ma, Kausik Sivakumar, Jason Yan, Osbert Bastani, and Dinesh Jayaraman. Learning policy-aware models for model-based reinforcement learning via transition occupancy matching. In *Proceedings of the 5th Annual Learning for Dynamics and Control Conference*, 2023. URL https://proceedings.mlr.press/v211/ma23a.html.

Ali Malik, Volodymyr Kuleshov, Jiaming Song, Danny Nemer, Harlan Seymour, and Stefano Ermon. Calibrated model-based deep reinforcement learning. In *Proceedings of the 36th International Conference on Machine Learning*, 2019. URL https://proceedings.mlr.press/v97/malik19a.html.

Andreas Maurer and Massimiliano Pontil. Empirical Bernstein bounds and sample variance penalization. In *Conference on Learning Theory (COLT)*, 2009.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems*, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/8558cb408c1d76621371888657d2eb1d-Abstract.html.

R. Tyrrell Rockafellar and Stanislav Uryasev. Optimization of conditional value-at-risk. *The Journal of Risk*, 2(3):21–41, 2000. doi: 10.21314/JOR.2000.038.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Rombach_High-Resolution_Image_Synthesis_With_Latent_Diffusion_Models_CVPR_2022_paper.html.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022. URL https://mlanthology.org/iclr/2022/salimans2022iclr-progressive/.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588 (7839):604–609, 2020.

Younggyo Seo, Danijar Hafner, Hao Liu, Fangchen Liu, Stephen James, Kimin Lee, and Pieter Abbeel. Masked world models for visual control. In *Proceedings of the 6th Conference on Robot Learning*, 2023. URL https://proceedings.mlr.press/v205/seo23a.html.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=St1giarCHLP.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021b. URL https://openreview.net/forum?id=PxTIG12RRHS.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, 2023.

Ruixiang Sun, Hongyu Zang, Xin Li, and Riashat Islam. Learning latent dynamic robust representations for world models. In *Proceedings of the 41st International Conference on Machine Learning*, 2024. URL https://proceedings.mlr.press/v235/sun24n.html.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, Timothy Lillicrap, and Martin Riedmiller. DeepMind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012.

Claas Voelcker, Marcel Hussing, Eric Eaton, Amir-massoud Farahmand, and Igor Gilitschenski. Calibrated value-aware model learning with probabilistic environment models. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025. arXiv:2505.22772.

Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. DayDreamer: World models for physical robot learning. In *Proceedings of the 6th Conference on Robot Learning*, 2023. URL https://proceedings.mlr.press/v205/wu23c.html.

Marvin Zhang, Sharad Vikram, Laura Smith, Pieter Abbeel, Matthew Johnson, and Sergey Levine. SOLAR: Deep structured representations for model-based reinforcement learning. In *Proceedings of the 36th International Conference on Machine Learning*, 2019. URL https://proceedings.mlr.press/v97/zhang19m.html.

APPENDIX CONTENTS

A The protocol in full 14
B Proofs 14
C Qualitative results 16
D Rollout matrices 19
E Error attribution in full 20
F Complete numerical results 20
F.1 Value-head blur control 23
G What an allocation policy is allowed to read 24
H Sensitivity to the protocol's free parameters 26
I Comparisons and controls 27
J Testing the mechanism at training time 27
K Closed loop, reported as a null result 30
L The structure of what is calibratable 30
M Additional practical controls 31
N Fidelity axes in the stored rollouts 31
O Statistical and metric robustness 32
P Additional results 35

A THE PROTOCOL IN FULL

Figures 1 and 2 give the procedure in the three stages the method runs in. Table 2 collects the key variables of §3, their meanings, and the stage that produces them. Each stored record in the result store carries the settings it was produced with, so the symbols below can be traced to concrete values experiment by experiment.

Table 2: **Key protocol symbols, grouped by the stage that produces them.**

Stage	Symbol	Meaning
Measurement (§3.2)	$s,\ a_k$	anchor state, restored exactly before every replay; candidate actions $k = 1, \dots, K$
	$P_{\text{ref}}(\cdot \mid s, a_k)$	simulator reference law, estimated from R replays under per-step actuator noise
	$Q_{\theta,S},\ Q_{\theta,\infty}$	the model's law under an S-step sampler and under the converged solve, both from the same initial noise z_0
	$\phi,\ o_k$	readout feature and per-candidate offset $\mathbb{E}_Q[\phi] - \mathbb{E}_{P_{\text{ref}}}[\phi]$, estimated on one half of the replays and scored on the other
Admissibility (§3.2)	$\text{sd}_{\text{replay}},\ \gamma$	replay-noise floor the conditional spread must clear; ranking identifiability $(Q_{(1)} - Q_{(2)})/\,\text{se}(Q) > 2$
Decomposition (§3.2)	ρ	dispersion ratio of model to reference; split into a learned-model and a finite-inference term by Eq. (3)
	$\bar{o},\ d_k$	candidate-shared and candidate-relative parts of the offset
	$\alpha,\ \beta$	the instrument that removes d_k (α) or rescales $\bar{o}$ (β) before the readout
Calibration (§3.3)	$T_\lambda,\ \lambda_{t,S,r}$	mean-preserving transport $m + \lambda(X - m)$; one scalar per task t, sampler budget S, and readout r, chosen on D_{fit}
	$\Delta_{t,S,r}$	paired gain on D_{gate}; retain the scalar only when the paired $2\,\text{se}$ screen and the seed-cluster 95% lower-bound veto both pass
Deployment	$\{\lambda_{t,S,r}\}$	accepted scalars are applied to held-out D_{test}; one lookup and one multiplication per value sample online

B PROOFS

Each statement of §3 is proved here and, independently, checked against Monte Carlo in `theory_check.py`, which prints a PASS line per claim. Throughout Q has mean m, covariance Σ and finite second moment, and $T_\lambda(X) = m + \lambda(X - m)$.

Proposition 2(i). T_λ is affine, so $\mathbb{E}[T_\lambda X] = m+\lambda(\mathbb{E}[X]-m) = m$ and $\mathrm{Cov}[T_\lambda X] = \mathrm{Cov}[\lambda X] = \lambda^2\Sigma$. Both are exact for every λ and require nothing of Q beyond a finite second moment; in particular the transport is mean-preserving by construction, and not only in expectation over a fit. □

Proposition 2(ii). With $r(z) = w^\top z + c$, $F_r(T_\lambda \# Q) = \mathbb{E}[w^\top(m + \lambda(X - m)) + c] = w^\top m + \lambda w^\top(\mathbb{E}[X] - m) + c = w^\top m + c$, independent of λ. □

This is a scope statement, not a technicality: a planner whose statistic is the expectation of an affine function of the sampled future cannot be helped *or* harmed by any mean-preserving dispersion transport. The readouts DARC does move are moved through curvature.

Proposition 3. Write $\delta = X - m$, so $\mathbb{E}[\delta] = 0$. A second-order expansion of r about m gives $r(m+\lambda\delta) = r(m)+\lambda\nabla r(m)^\top\delta+\frac{\lambda^2}{2}\delta^\top H_r(m)\delta+o(\lambda^2\|\delta\|^2)$. Taking expectations, the linear term vanishes because $\mathbb{E}[\delta] = 0$, and $\mathbb{E}[\delta^\top H_r(m)\delta] = \mathrm{tr}[H_r(m)\Sigma]$, which gives the stated expansion. When r is quadratic the remainder is identically zero and the expansion is exact. □

The consequence used in the main text is that the effect of the transport is governed by $\mathrm{tr}[H_r\Sigma]$, a coupling between the model's covariance and the readout's curvature, and not by any norm of the model's distributional error. Two models at equal energy distance from P_{ref} can therefore sit at very different decision errors, which is why we report an attribution in place of a single distance.

Proposition 4. By Proposition 2, $T_\lambda \# Q = \mathcal{N}(b, \lambda^2\sigma^2)$. For the latent objective, $\mathrm{KL}(\mathcal{N}(b, \lambda^2\sigma^2) \,\|\, \mathcal{N}(0,1)) = \frac{1}{2}(\lambda^2\sigma^2 + b^2 - 1 - \log\lambda^2\sigma^2)$, whose derivative $\lambda\sigma^2 - 1/\lambda$ vanishes at $\lambda^2\sigma^2 = 1$, giving $\lambda_{\mathrm{lat}} = 1/\sigma$ with no dependence on b. For the decision objective, $F_r(\mathcal{N}(\mu, v)) = \mu^2 + v$ when $r(z) = z^2$, so $F_r(T_\lambda \# Q) = b^2 + \lambda^2\sigma^2$ and $F_r(P_{\mathrm{ref}}) = 1$; the loss $|b^2 + \lambda^2\sigma^2 - 1|$ is zero at $\lambda_{\mathrm{dec}} = \sqrt{1 - b^2}/\sigma$ whenever $|b| \le 1$.

The two are not merely different. The decision loss is 0 at λ_{dec} and b^2 at λ_{lat}; the divergence is $b^2/2$ at λ_{lat} and $-\frac{1}{2}\log(1 - b^2)$ at λ_{dec}, and $-\log(1 - x) > x$ on $(0, 1)$. Both gaps are strict for every $b \neq 0$, so each optimum is strictly worse under the other's objective. They also fall on opposite sides of 1 whenever $\sigma < 1 < \sigma/\sqrt{1 - b^2}$, in which case a latent-fitted criterion asks for the correction to be applied in the wrong direction, and not merely at the wrong size. □

Proposition 1. By the fundamental theorem of calculus applied to $t \mapsto \mathbb{E}[r(X_i + tb)]$, $F_i(b) - F_i(0) = \int_0^1 \mathbb{E}[\nabla r(X_i + tb)]^\top b\, dt$, so the change in the gap between two candidates is exactly the $\Delta_{ij}(b)$ of §3.2. If r is affine then $\nabla r \equiv w$ and the bracket is identically zero, so $F_i - F_j$ is invariant to b and the argmax cannot move. If $r(z) = z^2$ and $X_i \sim \mathcal{N}(m_i, s_i^2)$ then $F_i(b) = (m_i + b)^2 + s_i^2$, so $F_1(b) - F_2(b) = (m_1 - m_2)(m_1 + m_2 + 2b) + s_1^2 - s_2^2$ is affine in b with slope $2(m_1 - m_2)$; when $m_1 \neq m_2$ it has the single root $b^\star$ stated, and crosses zero there. □

Proposition 2(ii) and Proposition 1 are two faces of one fact about affine readouts, applied to different perturbations: the first to a change of *scale*, the second to a change of *location*. This is the formal content of the claim that the dispersion ratio and the offset are different kinds of object and are not interchangeable targets for a calibration set.

Proposition 5. Write $\epsilon = \max_k \epsilon_k$. For any $j \neq k^*$, $\tilde{q}_{k^*} - \tilde{q}_j \ge q^*_{k^*} - q^*_j - \epsilon_{k^*} - \epsilon_j \ge \gamma - 2\epsilon$. The first claim follows when $\gamma > 2\epsilon$. Since $\tilde{q}_{\hat{k}} \ge \tilde{q}_{k^*}$, adding and subtracting these estimates gives $q^*_{k^*} - q^*_{\hat{k}} \le (q^*_{k^*} - \tilde{q}_{k^*}) + (\tilde{q}_{\hat{k}} - q^*_{\hat{k}}) \le \epsilon_{k^*} + \epsilon_{\hat{k}} \le 2\epsilon$. The inequality is a sufficient condition, not a claim that average estimation gains improve argmax choices. □

Proposition 6. The budget-constrained problem $\min_{s(\cdot)} \mathbb{E}_\mu[R(c, s(c))]$ subject to $\mathbb{E}_\mu[s(c)] \le B$ has, for the Lagrangian $L(s; \tau) = \mathbb{E}_\mu[R(c, s(c)) + \tau s(c)]$, the pointwise minimiser $s^\star(c) \in \arg\min_s\{R(c, s) + \tau s\}$, and $\tau(B)$ is chosen so the budget binds. The objective decouples across contexts because both the risk and the constraint are additive in c, so no coupling survives the minimisation. When the marginals are non-increasing in j, the pointwise minimiser is the largest rung whose marginal still clears τ, which is the threshold form. In either form, $R(c, s) \mapsto R(c, s) + a(c)$ shifts the objective by a constant in s and cannot move the minimiser. □

The substantive content is the last sentence. The quantity a planner would most naturally measure about a context, how badly the model does there, is exactly the quantity the optimal allocation is invariant to.

Theorem 7. By Proposition 6 the oracle policy is $s^\star(c) = \max\{s_j : g_j(c) \geq \tau\}$. A policy measurable with respect to h reproduces it for every budget if and only if it can reproduce each event $\{g_j \geq \tau\}$, which is the stated condition. For the second claim, Corollary 8 supplies a level-sufficient statistic with $V_h = 0$, so level sufficiency is not sufficient; and a statistic carrying g without information about the level attains the oracle while determining R only up to $a(c)$, so it is not necessary. □

Corollary 8. With $R(c, s) = a(c)+\psi(c, s)$ the marginals are $g_j(c) = [\psi(c, s_{j-1})-\psi(c, s_j)]/(s_j - s_{j-1})$, free of a. If h is a function of a alone and $a \perp \psi$, then $g \perp h$, so $\mathbb{E}[g \mid h] = \mathbb{E}[g]$ and every h-measurable policy is a constant allocation, whose value over the best fixed budget is zero. □

Proposition 9. Standardise g and $\hat{g}$ and let z_{1-p} be the $(1 - p)$-quantile of the standard normal. Selecting the top p by $\hat{g}$ and evaluating the realised marginal gives $\mathbb{E}[g \mid \hat{g} \geq z_{1-p}] = \rho\, \varphi(z_{1-p})/p$, because $g = \rho\hat{g} + \sqrt{1 - \rho^2}\,\epsilon$ with $\epsilon \perp \hat{g}$ and the second term contributes nothing under selection on $\hat{g}$. Selecting on g itself gives $\varphi(z_{1-p})/p$. The excess over the unselected mean is the value of the allocation in each case, so the ratio is ρ, and the factor $\varphi(z_{1-p})/p$ cancels. □

Two consequences are worth stating. The recovered fraction is linear in ρ, not quadratic, so a proxy correlating 0.3 with the marginal recovers 30% and not 9%; a method reporting a small recovered fraction is therefore reporting a very weak correlation. And the ratio does not depend on p, so a policy cannot be rescued by making the allocation sparser.

Scope of the allocation theory. Corollary 8 assumes the level and the shape vary independently. When they do not, a level statistic inherits some information about the marginal and the recovered value need not vanish. On one of our two attribution domains the true dispersion does correlate with the per-context optimum, at $+0.61$ against a permutation band of $[-0.19, +0.51]$, while on the other it sits inside the band at $+0.22$. Those correlations are computed over 24 cells and we draw no conclusion from them. The properly powered statement is Table 7, which measures recovered allocation value, not a correlation, and which finds the oracle level statistic no better than the learned proxies for it.

Gate coverage and scope. The acceptance rule $\Delta_r > 2\,\mathrm{se}(\Delta_r)$ is a one-sided screen on the paired gain over anchors held out of the fit. Its nominal level rests on a normal approximation that treats anchors as independent, which they are not: anchors within a task share a checkpoint, so the effective sample size is smaller than the anchor count and the nominal level is optimistic. A distribution-free alternative is available: for gains bounded in a known range, the empirical-Bernstein bound of Maurer & Pontil (2009) gives a one-sided lower confidence limit at the cost of a larger threshold and therefore fewer accepted cells. We did not adopt it because the operative evidence here is measured: over the main grid the gate admits 0 harmed cells of 156, and over the full readout sweep 1 of 1620. We state the rule as a screen, and we do not claim a distributional guarantee for it.

Definition 1 (Amortized calibratability)**.** For an error statistic $e(c)$, a deployment-visible summary $h(c)$ and a class $\mathcal{F}$, $\mathcal{C}_{\mathcal{F}}(e) = 1 - \inf_{f \in \mathcal{F}} \mathbb{E}\|e(c) - f(h(c))\|^2 / \mathbb{E}\|e(c)\|^2$.

C QUALITATIVE RESULTS

This appendix supplements §4.3. The six columns are fixed throughout: the simulator frame, the model's prediction, the prediction error between them, the component of that error which every candidate action shares, the component that differs between candidates, and the differing component laid over the frame. Every map in a block uses one colour scale, chosen per scene as the maximum over the three error maps, so panels within a block are directly comparable while absolute intensities are not comparable across scenes.

Three things are visible in these figures and are worth stating explicitly, because they are the qualitative counterpart of the numerical claims in the main text. First, prediction error is not spread uniformly: it

concentrates on the moving links, the gripper, contact regions and the cast shadow, which is where the conditional law is genuinely multi-modal. Second, the shared and differing components are spatially distinct, and neither is a scaled copy of the other. The shared component covers the whole articulated body, since every candidate action moves it in broadly the same way over a short horizon, whereas the differing component concentrates in the small region where the candidate actions actually diverge. Third, the feature-space α intervention of §3.2 tests the causal relevance of the differing component: removing it can move the decision. This diagnostic intervention is separate from DARC, which rescales value samples after the value head and leaves these image maps unchanged.

Scenes are ordered by measured readout gain, and both the strongest and the weakest are shown; we do not present a selection of favourable cases. The weakest scene is included precisely because its differing component is less sharply localised, which is the visual signature of the cases where the correction helps least.

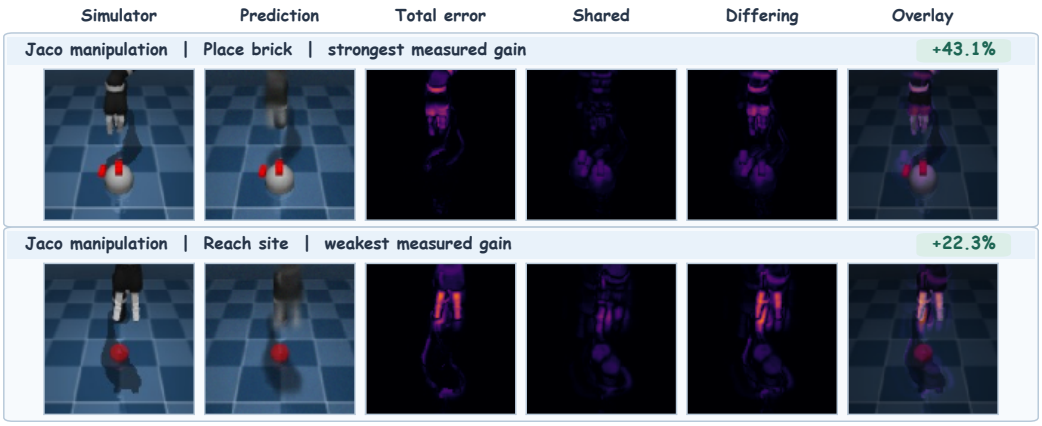


Figure 6: **Qualitative decomposition on the Jaco manipulation model.** Left to right: the simulator frame, the model's prediction, the prediction error, the part of that error every candidate action shares, the part that differs between them, and the differing part laid over the frame. Rows are the strongest and weakest scenes by measured readout gain. Error concentrates on the arm, the gripper and the manipulated object; the shared and differing parts are spatially distinct, and only the differing part changes when the candidate action changes. Locomotion scenes are in Figure 7, and four-step sequences for both families follow.

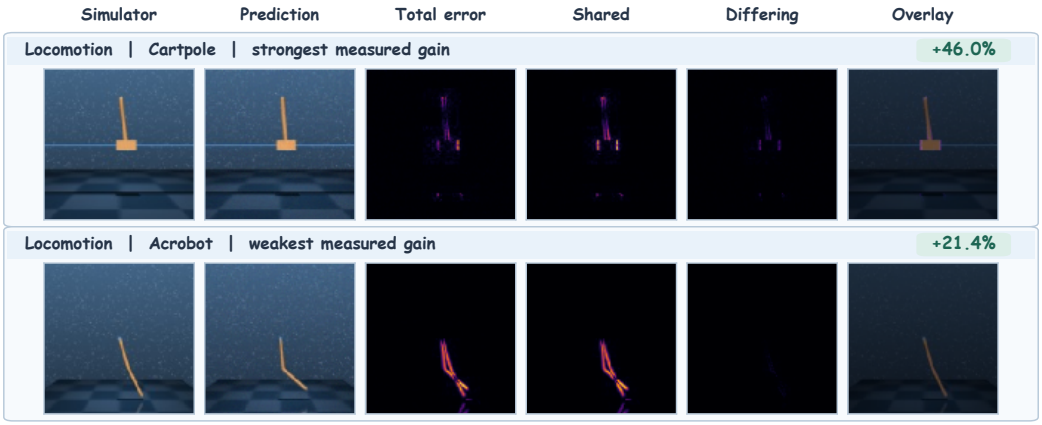


Figure 7: Qualitative decomposition on the nine-environment locomotion model, same columns as Figure 6.

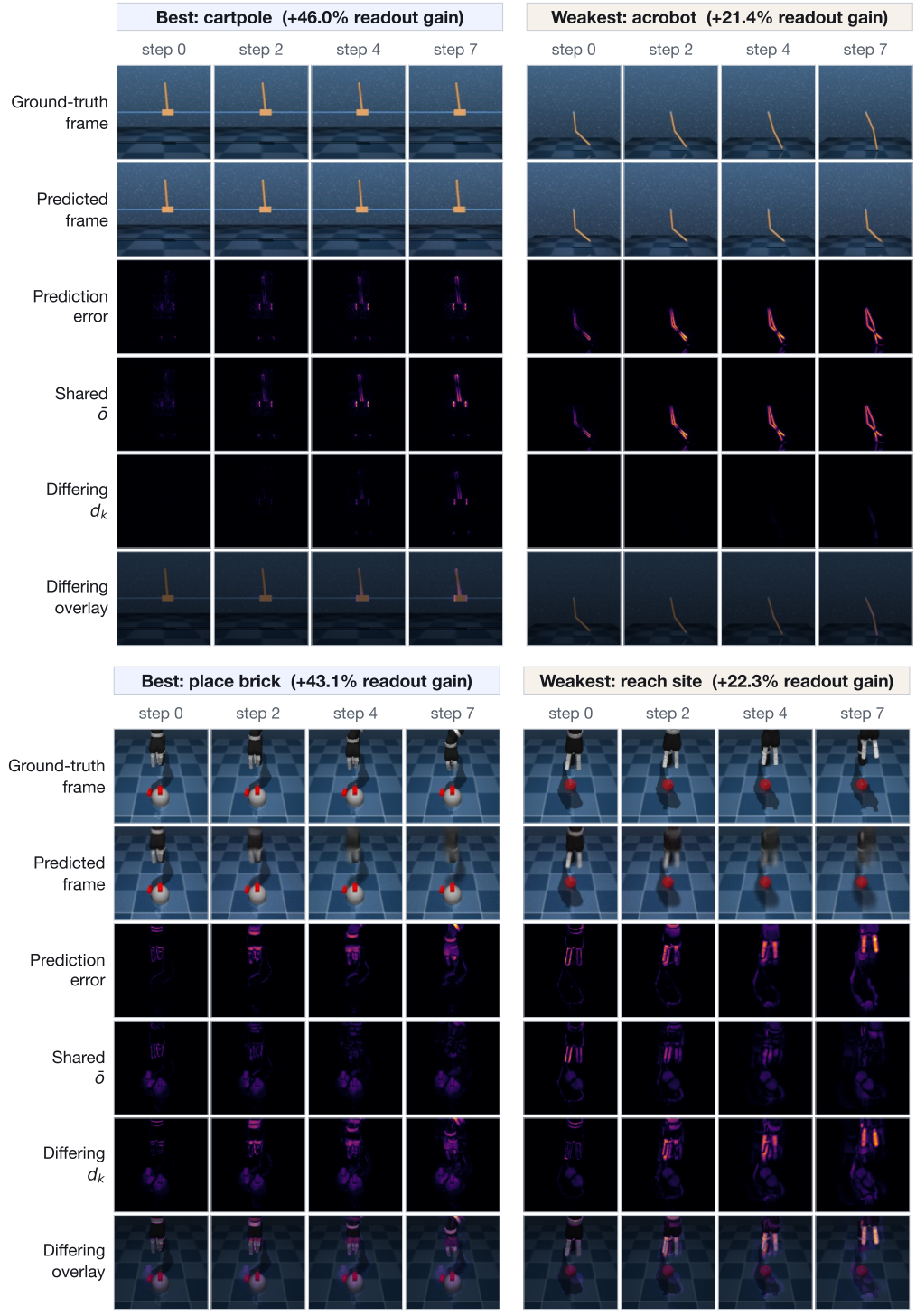


Figure 8: **Four horizon steps on both model families. Top:** the locomotion scenes with the largest and the smallest readout gain. **Bottom:** the same view on the Jaco manipulation model. Rows are the ground-truth frame, the prediction, the prediction error, the shared component $\bar{o}$, the differing component d_k and the differing overlay; columns are horizon steps. The differing component stays confined to the parts of the scene the candidate actions move, at every step.

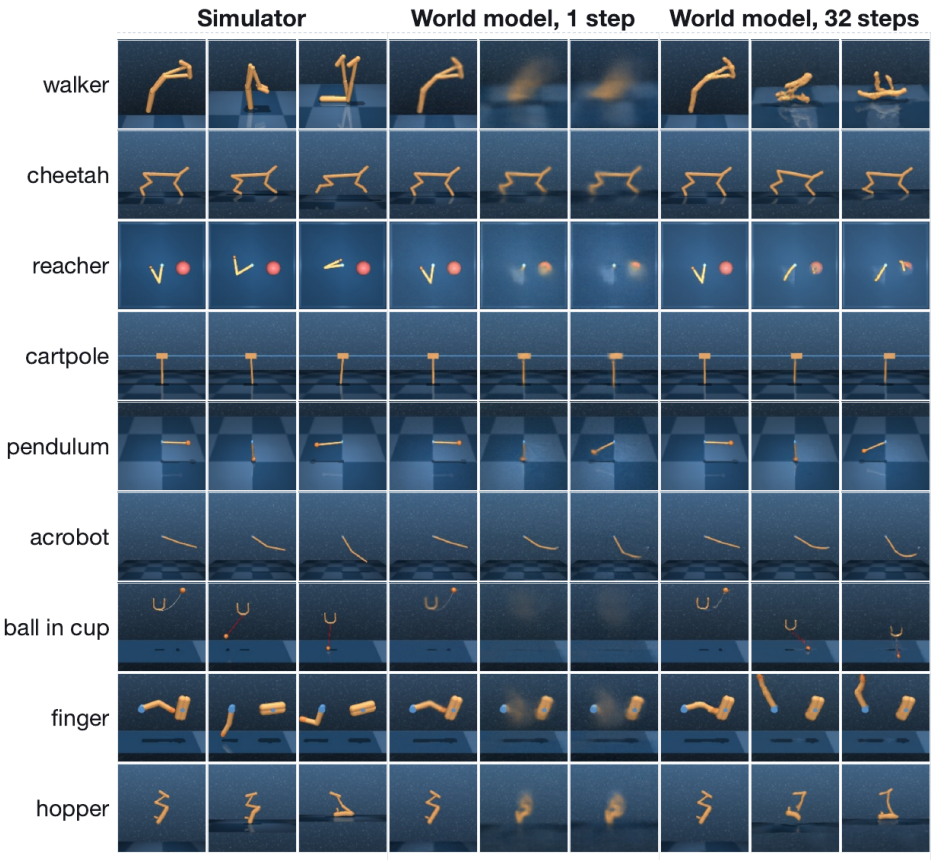


Figure 9: **Every locomotion scene, simulator reference and two sampler step counts.** Three frames of the same forty-step autoregressive rollout, started from identical reference frames and driven by identical actions, so any difference across a row is the model and not the scene. The thirty-two-step column is visibly sharper than the one-step column; §4.3 reports that its readout error is nevertheless worse.

D ROLLOUT MATRICES

The matrices below compare simulator replays with one- and thirty-two-step world-model rollouts in every scene. Within a row, each block starts from the same two reference frames and uses the same action sequence. Initial model noise is paired, so the model columns isolate the effect of solver steps rather than a change of scene or sampled noise. Three frames are drawn at equal spacing from the same forty-step autoregressive rollout.

These matrices are the direct visual evidence for the dissociation reported in §4.2. Reading across a row, the thirty-two-step column is clearly the most detailed: edges are crisper and texture is recovered where the one-step column is smooth. Reading the numbers instead, that same column is further from the future that actually occurred and yields a worse readout error. The one-step column is close to the conditional mean, which is why it looks smooth and scores well on distortion; the thirty-two-step column is a genuine sample, which is why it looks realistic and scores worse. DARC is absent from the pixel columns because its deployed transport acts only on value samples after rendering. For a fixed model seed and action, its generated frames are exactly the one-step frames.

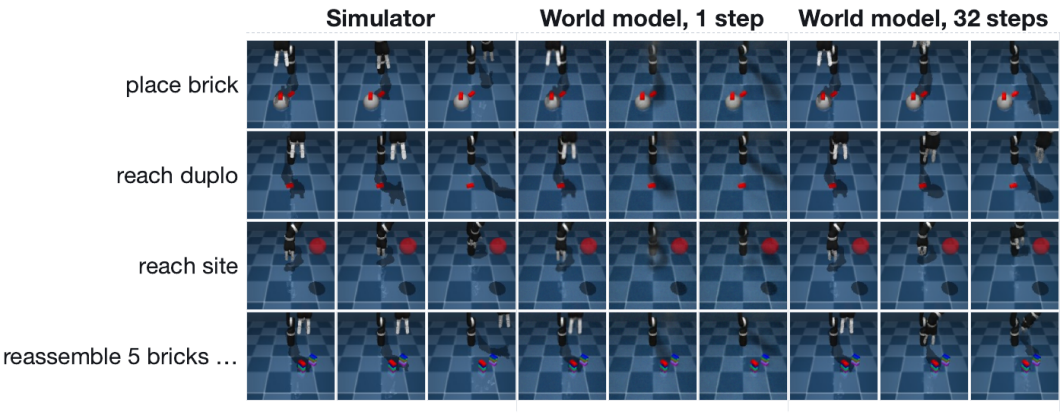


Figure 10: The four Jaco manipulation scenes under the same paired-noise protocol, from the second world model. DARC uses the one-step pixels shown here; its calibration is applied to subsequent value samples.

E Error attribution in full

This appendix supplements §4.2. The ladders are run with the initial noise held fixed, so the only thing that varies along them is how faithfully the same flow ODE is integrated. The attribution requires three checks before its two terms can be interpreted. First, the endpoint used as $Q_{\theta,\infty}$ must be converged for the statistic being decomposed. Second, the endpoint must be a property of the learned model rather than an artefact of one integration rule. Third, the reported learned-model share must remain above an equal split after uncertainty across cells is retained. Figure 11 reports these checks in that order. The Euler and Heun ladders approach the same scene-level limit, and the task-clustered intervals place the median learned-model share above one half in both tested scene families. These diagnostics license the provenance statement for the measured cells; they do not establish convergence for every pixel, task, horizon, or replay law.

F Complete numerical results

This appendix tabulates the aggregate results, selected cell-level comparisons, and sensitivity analyses used in the main text. The task, seed, sampler-step, and readout denominators are stated beside the corresponding comparisons so that the reported summaries can be interpreted directly.

Table 3 is the grid the headline number averages over. Two things are visible in it that an average hides: on every task where it is non-zero the gain shrinks as the step count grows, though not strictly (cartpole and reacher each rise once), and every refusal of the gate falls on pendulum, the task on which the correction has nothing to remove.

Table 3: **Energy distance removed by the gated spread calibration, per task and per sampler step count.** Entries are percentages; **bold** is a cell the paired gate accepted, grey a cell it declined. Averaged over tasks the gain falls monotonically in the step count (Figure 5c). The accepted pendulum cell at one step rounds to 0.0%.

Task	Sampler steps					
	1	2	4	8	16	32
cartpole	**7.2**	**4.7**	**4.8**	**3.9**	**2.8**	**2.9**
cheetah	**29.2**	**24.5**	**21.2**	**18.4**	**11.4**	**9.9**
pendulum	**0.0**	0.0	0.0	0.0	0.0	0.0
reacher	**22.0**	**18.2**	**21.6**	**18.4**	**15.4**	**13.0**
walker	**35.3**	**23.1**	**13.9**	**8.1**	**4.6**	**2.6**

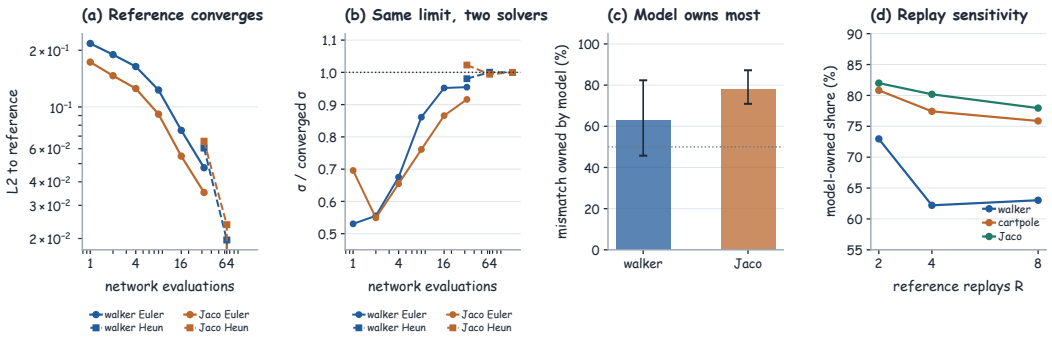


Figure 11: **The controls behind the attribution.** Equation (3) defines the attribution; this figure gives three checks that license it. The initial noise is held fixed throughout and the same flow ODE is integrated along every ladder. **(a)** The reference solve is converged for the statistic being attributed. Going from Heun 32 to Heun 64 moves the readout dispersion of the median cell by 0.61% on walker and 0.90% on the arm, worst cell 1.81% and 4.61%, which are 1.7% and 2.6% of the effect being attributed. We quote the per-cell figure because the per-cell changes carry mixed signs and an average over cells cancels them. **(b)** σ_∞ belongs to the learned model: normalised by each scene's own converged value, the Euler and the Heun ladders on both scene families approach the same limit, so naming $Q_{\theta,\infty}$ as the second reference law does not depend on which solver reaches it. **(c)** The learned-model term's measured share of the log-dispersion discrepancy: the median of each scene's 24 cells, with a 95% bootstrap interval that resamples those cells; both scene families sit above an equal split.

Table 4 tests spread and mean correction under a separate axis-control protocol. The spread scalar removes 17.50–33.21% on cheetah, reacher, and walker, and 0.00% on cartpole and pendulum. The best of three mean correctors removes at most 2.56% on any task. This variation supports task-specific calibration and prevents interpreting the spread direction as universal.

Table 4: **Spread and mean calibration controls, per task, at one sampler step.** Energy distance removed by a single offline spread scalar and by the best of three mean correctors (a global constant, ridge regression, or an MLP). The spread correction helps on three tasks, while the best mean correction removes at most 2.56% on any task. This separate axis diagnostic is not a cellwise decomposition of the gated gains in Table 3.

Task	Spread removed (%)	Best mean removed (%)
cartpole	0.00	0.65
cheetah	20.39	0.12
pendulum	0.00	0.70
reacher	17.50	2.56
walker	33.21	0.26

Table 5 compares held-out losses in cells where a context-specific oracle can improve over the plain prediction. The estimated per-context correction has a higher loss than the global constant in every listed cell; the oracle column quantifies the remaining attainable reduction.

Table 5: **Global and per-context corrections against an oracle.** Held-out readout loss (lower is better) on cells for which an oracle context-specific correction improves over the plain prediction. The recovery fraction for a fitted method is $(L_{\text{plain}} - L_{\text{method}})/(L_{\text{plain}} - L_{\text{oracle}})$; averaging this fraction over seven cells using unrounded losses gives 94.7% for the global constant and 30.7% for the encoder's per-context estimate. The oracle is independently estimated, so a small displayed reversal can occur.

Task	S	Plain loss	Global loss	Context loss	Oracle loss
cartpole	1	8.06	6.67	7.20	6.68
cheetah	1	11.60	6.74	10.26	6.63
cheetah	4	11.09	8.40	11.34	7.92
reacher	1	14.08	10.36	13.48	10.33
reacher	4	15.94	12.68	15.10	12.06
walker	1	19.64	10.22	15.42	10.15
walker	4	20.90	10.60	15.42	10.24

Table 6 is the invariance check. If the composition of the error were a property of the model and not of the task, it would move as the model is made better; across a factor of two to twenty in energy distance within each task, it does not.

Table 6: **The decomposition varies less with model scale than with task.** 81 trained models per task, spanning eight times the data, eight times the width and sixteen times the training steps. The energy-distance span is the ratio of worst to best model within a task. Differing share is reported as mean $\pm$ standard deviation across 81 models; its largest within-task standard deviation is 4.7 percentage points, versus a 58.1-point difference between the highest and lowest task means.

Task	Models	Energy span ($\times$)	Differing share (%)
cheetah	81	2.66	73.3 ± 1.0
reacher	81	3.57	60.6 ± 3.2
walker	81	2.29	56.7 ± 4.7
cartpole	81	5.46	51.1 ± 0.6
pendulum	81	20.49	15.2 ± 0.4

Table 7 is the full sufficiency sweep behind Figure 12. The two curve-reading rows climb with the budget and the four scalar rows do not, which is the whole of the argument that a per-context step count is not worth predicting from any scalar summary.

Table 7: **Value recovered by each allocation statistic, by bin budget.** The table reports the percentage of the improvement available to a per-context allocation policy, against the number of bins the policy may use. Reading the risk curve directly, or its slope, climbs with the budget. Every per-candidate *scalar*, including the true spread supplied as an oracle, stays inside a narrow band at every budget, against a permutation null of $[-3.8, -0.0]$. The ceiling belongs to the statistic, not to the estimator.

Statistic	Bins available to the policy			
	2	3	5	8
Direct observation of the risk curve	38.1	52.1	80.9	87.9
The slope of that curve	42.8	50.7	58.5	61.7
The model's own spread	7.2	6.6	6.1	6.5
A confidence head	5.1	5.9	6.1	5.9
Generator-to-teacher divergence (the DSA criterion)	4.9	3.8	4.5	4.2
The true spread, supplied as an oracle	5.4	6.1	7.7	7.3

Table 8 prices the two currencies against each other. The cost column is measured wall clock on one machine with the encoder included, normalised to a single plain sampler step.

Table 8: **Decision benefit and measured cost of test-time computation.** Decision error removed, averaged over 45 task × seed × budget cells, with measured wall-clock cost relative to a single plain step. Adding 32 model samples yields only a 0.4-point change in this comparison.

Intervention	Error removed (pts)	Relative cost
32× more model samples	0.4	32.0×
shared-part correction	0.6	1.0×
one real rollout each	29.9	1.4×
six real rollouts each	45.9	2.6×
model-based, budget 1	26.8	1.4×
model-based, budget 6	38.2	2.6×

Table 9 is the matched-compute comparison. The router is an implemented per-frame scheduler run at its own operating point; the calibrated sampler is run at four steps, which is fewer steps than the router chooses on average.

Table 9: **Matched compute against a per-frame confidence router.** ED is energy distance (lower is better). The router runs at its measured operating point; the calibrated sampler uses four steps, slightly fewer than the router's 4.3-step mean.

Task	Router ED	Router steps	DARC ED	Improvement (%)
cheetah	11.04	4.3	8.27	25.1
reacher	15.09	4.3	12.25	18.9
walker	20.56	4.3	16.09	21.8

Table 10 reports the closed-loop measurements in full, including the arms that do not separate. At this sample size the return shows no trend in the step count, and the calibration is not shown to improve control.

Table 10: **Closed loop, reported as a null result.** Left: return above a random policy against the sampler step count. Right: `PointMaze_UMaze-v3` success rate. The return shows no trend in the step count at this sample size, and the calibrated and uncalibrated arms do not separate. We do not claim that readout calibration improves closed-loop control.

Task	Arm	Gain (%)	Gain/s.e.	n
cheetah	model $s = 1$	8.4	0.9	100
cheetah	model $s = 4$	-4.8	0.6	100
cheetah	model $s = 8$	7.0	0.8	100
cheetah	model $s = 32$	10.3	1.2	100
walker	model $s = 1$	26.2	4.1	100
walker	model $s = 4$	32.4	4.7	100
walker	model $s = 8$	33.6	4.5	100
walker	model $s = 32$	24.9	4.0	100

Arm	Success (%)	s.e. (pts)	n
random	28.3	5.8	60
model s=4	43.3	6.4	60
model s=32	40.0	6.3	60
DARC s=1	33.3	6.1	60
DARC s=4	41.7	6.4	60

Table 11 breaks the video world model down by task, including the three tasks whose value head does not fit well enough to carry a readout. Those rows are dropped from the aggregate and listed here; none is omitted silently.

F.1 Value-head blur control

The value head fits real-frame return-to-go at $R^2 = 0.998$. We blur real frames with known returns to check whether degradation at one sampler step creates a spurious head bias. Severe blur ($\sigma = 3$) reduces the fit to $R^2 = 0.20$. Model samples lie in the equivalent $\sigma = 0.75$–0.99 band. There, bias changes only from -0.0243 to -0.0250 (0.0008 normalised units), while R^2 remains 0.985–0.990. Residual standard deviation rises from 0.093 to 0.120, so the blurrier one-step samples carry noise

Table 11: **Nine-environment pixel-space model at one sampler step.** R^2 is the value head's fit. Rows with $R^2 < 0.5$ (reacher, ball_in_cup, hopper) retain their diagnostic values here but are excluded from the main aggregate. ρ is the model-to-reference dispersion ratio; the candidate column is the candidate-dependent offset share. Monotone counts how many of four readouts have a monotone response in the intervention sweep; harmed is the number of worsened cells out of 288 per task under the paper's 0.5% threshold.

Task	R^2	ρ	Candidate (%)	Gain (%)	Monotone	Harmed
walker	0.998	0.18	87.4	23.7	4/4	87
cheetah	0.974	1.70	93.7	13.6	4/4	40
reacher	0.000	0.99	93.1	56.1	0/4	14
cartpole	0.774	32.90	70.3	45.8	4/4	15
pendulum	0.856	29.11	32.2	44.8	0/4	23
acrobot	0.995	93.52	12.0	22.5	4/4	0
ball_in_cup	0.011	0.30	90.8	40.6	2/4	42
finger	0.602	0.32	91.6	8.5	3/4	37
hopper	0.000	1.65	81.1	0.0	4/4	0

Table 12: **Scene-level results for the two pixel-space model families.** Readout error removed is averaged over three sampler step counts and four readouts (higher is better), with relative gain clipped below at -200% per cell. Harmed counts record a paired readout-error increase greater than 0.5%; each row contains 864 cells. R^2 is value-head fit. The three locomotion scenes with $R^2 < 0.5$ are omitted from this aggregate and shown in Table 11.

			Error removed (%)		Cells harmed		
Family	Scene	R^2	Readout	Latent	Readout	Latent	Cells
locomotion	walker	0.998	34.6	-20.5	148	269	864
	cheetah	0.974	32.4	27.7	107	266	864
	cartpole	0.774	46.0	48.2	39	111	864
	pendulum	0.856	45.1	34.6	86	202	864
	acrobot	0.995	21.4	27.8	15	223	864
	finger	0.602	30.3	0.1	115	265	864
Jaco	reassemble bricks	0.999	42.8	-10.4	181	247	864
	place brick	0.999	43.1	-16.2	112	362	864
	reach site	0.959	22.3	-8.6	65	264	864
	reach duplo	0.996	32.3	-5.8	119	256	864

adverse to DARC. Blur is a proxy for sample degradation; this control addresses differential head bias, while the simulator reference still passes through the same head.

Table 12 gives the scene-level values behind the family aggregates and retains both the readout-space and latent-space criteria with their paired harm counts.

G What an allocation policy is allowed to read

This appendix supplements §3.5. A planner on a fixed budget would like to spend sampler steps where they help most. Every method that attempts this predicts, from some summary of the context, how much compute that context should receive. Four summaries we tested recover between 3.8 and 7.7% of the value a per-context allocation has available, against up to 87.9% for reading the risk curve itself. One of the four is the model's *true* dispersion at that context, handed to the allocator as an oracle, and it does no better than a confidence head trained to guess it. A failure an oracle does not repair is not an estimation failure; it is a statement about the object being estimated.

Marginal-return condition. Keep moving a step from one context to another for as long as it buys more where it arrives than it gave up where it left. At the optimum no such move remains, so the *marginal* return of a step is the same everywhere. Note what that condition never mentions. A context can be uniformly terrible and deserve no extra compute, if a step there buys nothing; a

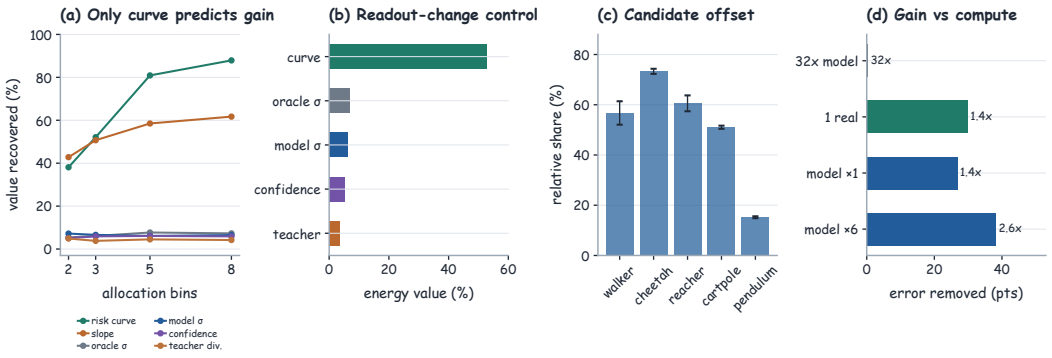


Figure 12: **Allocation signals and the cost of acquiring them.** **(a)** The risk curve and its slope recover substantially more decision value than four scalar proxies across tested bin budgets. **(b)** The same ordering holds when the readout is energy distance: direct curve observation recovers 52.7%, and no scalar exceeds 7.0%. **(c)** The candidate-relative offset carries a task-dependent share of total error; error bars span variation across trained models. **(d)** At measured wall-clock cost, a real rollout removes 29.9 readout-error points at 1.4× cost, while 32× more model samples remove 0.4 points. The panels compare information available at different costs, not an implemented controller.

context can be easy and deserve a great deal, if one step is what resolves it. How bad a context is and how much a step helps there are different quantities, and the budget responds only to the second. Formally, fix a budget B, let $R(c, s)$ be the risk at context c under s steps, let a policy choose $s(c)$ with $\mathbb{E}_\mu[s(c)] \le B$, and write $g_j(c) = [R(c, s_{j-1}) - R(c, s_j)]/(s_j - s_{j-1})$ for the return on the j-th increment.

Proposition 6 (The optimum sees only the marginal)**.** *There is a multiplier $\tau(B) \ge 0$ with $s^\star(c) \in \arg\min_s\{R(c, s)+\tau(B)\, s\}$, which under diminishing returns is the threshold rule $s^\star(c) = \max\{s_j : g_j(c) \ge \tau(B)\}$. Either form depends on the context only through $g(c, \cdot)$, so replacing $R(c, \cdot)$ by $R(c, \cdot) + a(c)$ for any a leaves the optimal allocation unchanged.*

The last clause carries the argument: shifting a context's whole risk curve up or down moves the optimal allocation not at all.

Theorem 7 (What a policy must be able to read)**.** *A statistic h attains the oracle value at every budget if and only if $h(c)$ determines the event $\{g_j(c) \ge \tau\}$ for every increment j and threshold τ. Knowing the risk level $R(c, \cdot)$ is neither necessary nor sufficient for this.*

Corollary 8 (An oracle for the level is worth nothing)**.** *Write $R(c, s) = a(c) + \psi(c, s)$ with level a and shape ψ varying independently. A statistic revealing $a(c)$ exactly, and nothing about ψ, recovers zero value at every budget.*

The claim is stronger than it first reads: a level statistic fails in a way better estimation cannot repair, because what it estimates perfectly is orthogonal to the decision at hand.

Proposition 9 (What a partial signal is worth)**.** *Suppose a budget awards one extra increment to a fraction p of contexts, and that g and its best prediction $\hat{g} = \mathbb{E}[g \mid h]$ are jointly Gaussian with correlation ρ. Allocating by $\hat{g}$ then recovers $V_h/V^\star = \rho$ of the oracle value, at every p.*

The fraction is linear in the correlation, so a signal correlating 0.3 with the marginal return is worth 30% of the oracle and not 9%; single-digit recovery therefore reports a correlation near zero. It is also free of p, so a weak policy cannot be rescued by allocating more selectively.

Allocation value of scalar proxies. Every summary in our sufficiency table describes where the risk curve sits: dispersion reports how wide the model's predictions are, a confidence head how wrong it expects to be, a generator-to-teacher divergence how far it has drifted. None reports how fast error would fall given one more step. Corollary 8 therefore predicts all four are worth little, and predicts specifically that the oracle is worth no more than the learned ones. Table 7 measures that:

the four level statistics recover 3.8 to 7.7%, the oracle among them and not above them, while the two statistics reading the curve or its slope recover 38.1 to 87.9%.

H SENSITIVITY TO THE PROTOCOL'S FREE PARAMETERS

The protocol fixes a replay count R, a candidate count K and a Monte Carlo count n per candidate. The attribution of §4.2 uses $R = 8$, $K = 4$, $n = 24$. This appendix varies each of the three around that point, one at a time, on two locomotion tasks and one manipulation task, and reports what moves. The horizon H is set at training time and a sweep over it would require retraining, so it remains untested.

Two quantities have to be separated when reading the table. The attribution itself is the share of the dispersion mismatch carried by the learned model, and that is what the paper's claims rest on. The absolute dispersion ratio $\rho_\infty = \sigma_\infty / \sigma_P$ is a different quantity, and its denominator is a sample standard deviation over R replays, which is biased low when R is small.

The share is stable. Over the region $R \geq 4$, $n \geq 12$ and every K, it moves by at most 0.082 within a task, against 0.149 between the three tasks, so the parameters move it about half as far as the task does. Including $R = 2$ and $n = 6$ widens the within-task range to 0.204, and both of those settings are severely under-sampled. The two sign-level findings survive everywhere: the converged model is over-dispersed in at least 79% of cells at every setting, and the finite-inference term carries the opposite sign in at least 62% of them.

The absolute ratio is not converged in R. Going from $R = 4$ to $R = 8$ still moves ρ_∞ by up to 43%, and the direction is downward in all three scenes, consistent with downward bias in the sample reference dispersion at small R. We therefore use ρ_∞ as a qualitative indicator of over-dispersion and do not interpret its absolute magnitude as converged. The estimated attribution share also depends on the reference denominator, although its variation over the better-sampled settings is smaller than the across-task variation reported above.

Reading the sweep. Raw counts in the over-dispersed and opposing-sign columns scale with K, so comparisons across candidate counts use fractions or the share column. Changing n alters Monte Carlo precision for the model law without adding simulator replays; changing R alters the reference denominator itself. The split-half signal occasionally exceeds one under finite-sample estimation, so it is a diagnostic rather than a calibrated probability. These separate effects explain why the sign-level observations are more stable than the absolute ratio and why each estimate must retain its replay setting.

Table 13: **Varying the protocol's free parameters one at a time.** $R/K/n$ gives the setting; the submitted operating point is $8/4/24$. Share is the median share of the dispersion mismatch carried by the learned model. Over and opposite count the cells where the converged model is over-dispersed and where the finite-inference term has the opposite sign. Signal is a split-half estimate of the fraction of the measured offset that is not replay noise, computed from disjoint halves of the replays. "Reassemble bricks" abbreviates the reassemble-five-bricks task.

Task	Varies	$R/K/n$	Cells	Share	ρ_∞	Over	Opposite	Signal
cartpole	–	8/4/24	24	0.759	11.53	24/24	24/24	1.00
cartpole	R	2/4/24	24	0.808	33.03	24/24	24/24	1.00
cartpole	R	4/4/24	24	0.774	20.35	24/24	24/24	0.99
cartpole	K	8/2/24	12	0.775	12.07	12/12	12/12	0.99
cartpole	K	8/3/24	18	0.759	12.07	18/18	18/18	1.00
cartpole	K	8/6/24	36	0.731	11.43	36/36	36/36	1.01
cartpole	n	8/4/6	24	0.723	9.16	24/24	22/24	1.00
cartpole	n	8/4/12	24	0.752	12.28	24/24	22/24	1.00
cartpole	n	8/4/48	24	0.765	11.75	24/24	24/24	1.00
reassemble bricks	–	8/4/24	24	0.779	3.91	19/24	19/24	0.55
reassemble bricks	R	2/4/24	24	0.820	8.39	24/24	22/24	0.93
reassemble bricks	R	4/4/24	24	0.802	5.43	24/24	22/24	1.06
reassemble bricks	K	8/2/24	12	0.734	5.91	12/12	12/12	0.97
reassemble bricks	K	8/3/24	18	0.759	5.48	18/18	17/18	0.97

Table 13 (continued)

Task	Varies	$R/K/n$	Cells	Share	ρ_∞	Over	Opposite	Signal
reassemble bricks	K	8/6/24	36	0.779	4.23	31/36	31/36	1.00
reassemble bricks	n	8/4/6	24	0.810	3.77	19/24	15/24	0.83
reassemble bricks	n	8/4/12	24	0.786	3.81	19/24	17/24	0.50
reassemble bricks	n	8/4/48	24	0.806	3.88	19/24	16/24	0.60
walker	–	8/4/24	24	0.630	2.99	24/24	22/24	0.94
walker	R	2/4/24	24	0.730	6.88	24/24	22/24	0.60
walker	R	4/4/24	24	0.622	3.15	23/24	21/24	1.02
walker	K	8/2/24	12	0.574	2.38	12/12	11/12	0.89
walker	K	8/3/24	18	0.656	2.69	18/18	16/18	0.90
walker	K	8/6/24	36	0.630	3.13	35/36	32/36	0.95
walker	n	8/4/6	24	0.526	2.67	22/24	20/24	0.94
walker	n	8/4/12	24	0.605	2.57	24/24	19/24	0.98
walker	n	8/4/48	24	0.614	3.17	24/24	23/24	0.92

I Comparisons and controls

This appendix reports the three comparisons a reader is most likely to ask for: against an implemented per-frame scheduler at matched compute, against the same allocation idea carried out on a real autoregressive video model, and against generic versions of our own correction.

The first two are the comparisons that matter for positioning. A reader who has seen the per-frame scheduling literature will reasonably ask whether the measurement in this paper is simply a worse version of an allocation policy, so we implemented one and ran it at matched compute, in place of quoting its published numbers, and we then took the allocation signal that Le et al. (2026) use, the divergence of the generator from its distillation teacher, and ran it as a per-frame rule on a real autoregressive video model, where it and two other signals each land below what a single well-chosen fixed step count achieves. That is not a claim that adaptive allocation cannot work; it is the observation that on these models the heterogeneity such a policy would exploit is small enough that the policy's own estimation error consumes it.

The third comparison guards against a different objection. A correction that removes error might do so for reasons unrelated to the decomposition it was derived from: because any perturbation of the right magnitude helps, or because the constant is picking up something generic about the model family, and nothing about the task. Both are tested directly: a matched-norm random direction and another task's fitted constant both make the energy distance substantially worse, so the constant is carrying task-specific information, and is not acting as a generic regulariser.

J Testing the mechanism at training time

The account behind DARC is that the correction works because the learned model is over-dispersed and the fit repairs that. Contribution (3) in §1 states the falsifiable form: reduce the over-dispersion and the benefit should shrink. We report three tests. The first, a training-time penalty, failed, and we keep it because it identified what a penalty has to act on; the second removes the over-dispersion at inference time; the third removes it at training time, paying the cost the first one identified.

A training penalty without a rollout effect. We retrained the nine-environment model with the published recipe and one addition, a penalty $\lambda(\mathbb{E}\|\hat{v}\|^2/\mathbb{E}\|v\|^2 - 1)^2$ on the ratio of second moments of the predicted and target velocities, at $\lambda = 0.5$ and with every other hyper-parameter fixed. The intended effect was to suppress the surplus velocity variance that becomes sampled over-dispersion. It did the opposite: over 9 task-by-step cells the mean $|\rho - 1|$ rose from 6.3 to 7.4 and DARC's gain was unchanged at +47.2% against +46.5%.

The sign was wrong for a reason worth recording. Flow matching regresses $v = x_1 - x_0$ on z_t, and the Bayes predictor is $\mathbb{E}[v \mid z_t]$, whose second moment is *below* that of v by the law of total variance. In a well-fit model the ratio the penalty drives toward one is therefore already below one, so the penalty raises the predicted velocity's second moment and with it the sampled dispersion. A penalty that targets sampled dispersion has to integrate the sampler during training, which is what we were

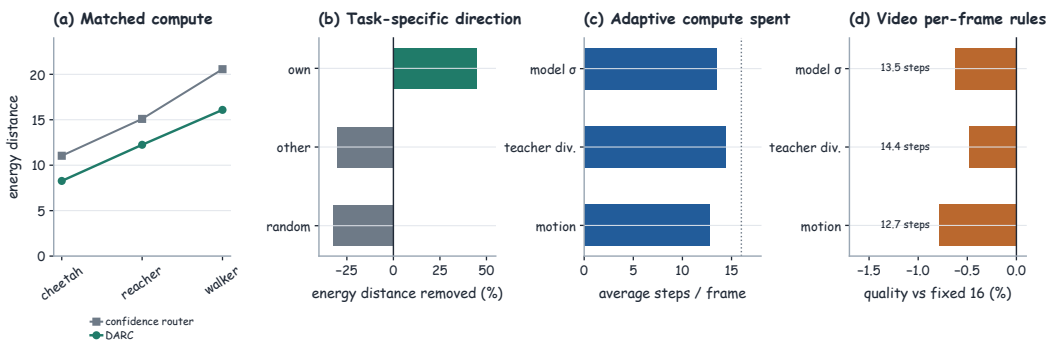


Figure 13: **Four controls for the proposed correction.** **(a)** At matched compute, DARC at four steps yields lower energy distance than a trained confidence router operating at 4.3 average steps on three tasks. **(b)** An own-task correction removes error, whereas transferred and matched-norm random corrections increase it. **(c,d)** On a real video model, three per-frame rules spend 12.7–14.4 average steps (c) and all score below the best fixed count of 16 (d). The two panels separate compute spent from resulting quality.

trying to avoid. That is exactly what the third attempt below pays for, and it is why this failure is worth recording: it located the cost that had to be accepted.

Inference-time de-dispersion. A cheaper instantiation applies the latent-variance scalar first, which drives ρ toward one without retraining, and then fits the readout scalar on the already corrected samples. If the benefit came from repairing over-dispersion, the second step should have little left to do.

The comparison is only a test where the model was over-dispersed to begin with, so we split the cells on $\rho > 1.3$, a property of the model that is fixed before any readout is scored. On the 12 over-dispersed cells across 5 tasks, cell-pooled direct and post-dedispersion incremental gains are +44.1% and +17.8%. The equal-task paired contrast is $+23.2 \pm 8.4$ points; these summaries use different weights, so the contrast is reported independently. On the 3 cells that were not over-dispersed the same comparison gives +37.0% and +68.1%, so the drop is absent exactly where the mechanism says it should be.

Two checks bound the reading. A drop of this kind could be manufactured if the composed arm were near a ceiling, since then a large first step would leave little room for a second. It is not: the correlation between the first step's gain and the second step's increment is -0.48 against a permutation band of $[-0.63, +0.62]$, where a ceiling would force a value near -1. And one task dissents. On reacher the model is over-dispersed at $\rho = 2.6$ and 4.5, the latent step does not help, and the readout step still delivers everything, for a drop of -2.2 points. The reacher row is retained; its atypical effect widens the task-clustered interval.

Training-time intervention. The first attempt failed because a penalty on velocity second moments does not act on the dispersion of integrated samples. The fix is to pay for what that diagnosis demands: integrate the sampler during training. We fine-tune the frozen checkpoint with the original flow-matching objective intact and one added term, the squared log-ratio between the dispersion of a few-step differentiable Euler rollout and the dispersion of the stored simulator replays at the same anchor, averaged over eight contexts per step, with an exponential moving average of the weights. Keeping L_{fm} preserves the original modeling objective. An earlier version that optimized dispersion alone degraded the other measurements.

The intervention does what the first attempt could not. Over the context bank the median $\sigma_{\mathrm{model}}/\sigma_{\mathrm{sim}}$ falls from 3.56 to 1.02 while the flow-matching loss stays at its own scale (0.0039 to 0.0105). Evaluating both checkpoints with the same script, seed and stored probes over 15 task-by-step cells gives the comparison the prediction asks for.

Table 14: The training-side test. Cells are grouped by whether the fine-tune moved the model *toward* calibration, which is a property of the intervention and is read off before any readout is scored.

Cells	Count	Change in DARC gain	Sign
$\lvert \log \rho \rvert$ fell, the intervention worked	5	**-9.5 points**	5 / 5 negative
$\lvert \log \rho \rvert$ did not fall	10	+0.7 points	mixed
Difference		-10.2 points	permutation p =0.10

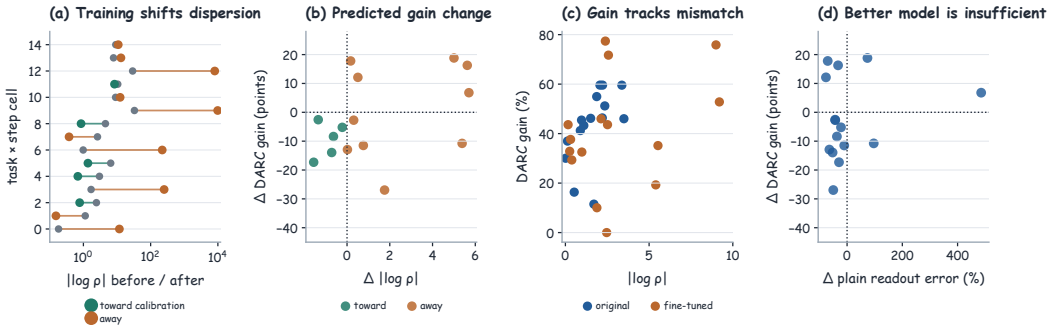


Figure 14: **The training-side test of the mechanism.** All four panels use the same 15 task-by-step cells. **(a)** What the fine-tune did to each of the 15 task-by-step cells, on the quantity that defines the grouping: $\lvert \log \rho \rvert$, the distance between the model's sampled dispersion and the simulator's, where 0 is a calibrated model. Arrows run from the original checkpoint to the fine-tuned one. The intervention moved 5 cells toward calibration and 10 away, and the split is read off here, before any readout is scored. **(b)** The prediction. Cells that moved toward calibration lost DARC gain without exception; cells that did not moved both ways. **(c)** The same prediction pooled over both checkpoints, one point per checkpoint-by-cell pair: the gain rises with how far the model sits from calibration. **(d)** The competing explanation is a change in uncorrected error. The gain change correlates more strongly with $\lvert \log \rho \rvert$ than with plain-error change; pendulum provides cells where the two predictions diverge.

Figure 14 and Table 14 report it. Where the fine-tune moved ρ toward one, DARC's gain fell in every cell, by -2.6 to -17.3 points and -9.5 on average; where it did not, the gain moved both ways and averaged +0.7. The group difference is -10.2 points at a permutation p of 0.10, which is suggestive on five cells and no more. The pooled form of the same prediction is sharper. Treating each checkpoint-by-cell pair as an observation, DARC's gain rises with how far the model sits from calibration: r =+0.316 over 30 observations, with a task-clustered permutation p of 0.026. Near-calibrated observations ($\lvert \log \rho \rvert < 1$) average 34.6% against 45.5% for the clearly mis-dispersed.

One confound has to be excluded, because the fine-tuned model is also simply better: its uncorrected readout error falls in most cells, which would leave DARC less to repair whatever the reason. The data do not support that explanation. The change in gain correlates +0.197 with the change in uncorrected error and +0.424 with the change in $\lvert \log \rho \rvert$, and the decisive cases are the ones where the two come apart: on pendulum the uncorrected error falls by 76.5% and 70.1% while DARC's gain *rises* by +12.1 and +17.8 points, because ρ there did not move toward one. In these cells the gain follows the over-dispersion, not the error.

We record two costs. The fine-tune was fitted at two integration steps and is out of that regime at one. At a single step ρ becomes extreme, so those cells fall outside the fine-tune's training regime. And the corrected model makes DARC harmful in more cells than the original (547 against 379 of 4320), which is the expected cost of pushing a model toward calibration with a term that sees only dispersion. The intervention therefore supports the predicted direction, in every cell where it reduced the over-dispersion and in the pooled association; the size of the group contrast rests on five cells and is not established.

Mechanism summary. Two independent manipulations of the over-dispersion, one at inference time and one at training time, move DARC's gain in the direction the account predicts. The inference-time test supports it on four of five tasks and fails on reacher. The training-time test supports it in all 5 cells where the fine-tune reduced the over-dispersion and in the pooled association. The group contrast has permutation $p = 0.10$; the fine-tune was fitted at two integration steps and increased the number of harmed cells. This is evidence for the mechanism, not a proof of it.

K CLOSED LOOP, REPORTED AS A NULL RESULT

The calibration's established contribution is to the estimates a planner consumes, not to the controller built on them. We report what we measured in closed loop, and we report it as a null result; it is not a benefit.

A readout improvement can change control only when it changes a comparison near an action margin and when the downstream controller visits states where that change is useful. The candidate-choice audit in §4.6 already shows that lower average readout error did not improve the measured ordering endpoints. The closed-loop evaluation adds the remaining feedback effects: actions alter the next state, the candidate set is rebuilt, and model error accumulates along the visited trajectory. We therefore keep return and success separate from readout error rather than using either as a proxy for the other.

Walker and cheetah use 100 episodes per arm at each tested sampler budget. PointMaze uses 60 episodes per arm at four steps and reports success rather than return. The PointMaze means are $43.3 \pm 6.4\%$ for the plain arm and $41.7 \pm 6.4\%$ for DARC; their uncertainty intervals overlap. Walker has no consistent sampler-step trend, and cheetah provides no clear calibrated advantage. Panel (d) measures whether the plain model-policy arms differ from a random controller; it is a sanity check on the task signal and is not evidence for DARC. These experiments establish that the controller can be evaluated with the calibrated statistic, while leaving the control-benefit hypothesis unresolved.

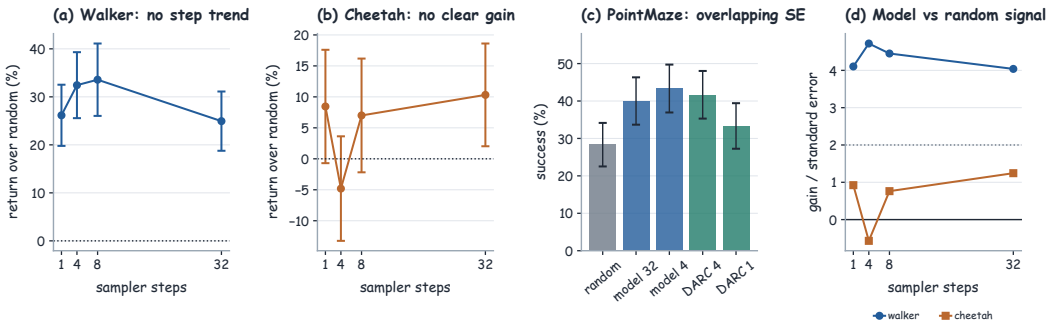


Figure 15: **Closed-loop evidence does not establish calibration benefit. (a,b)** Walker and cheetah model-policy return over random across sampler steps (100 episodes per arm, bars ± 1 standard error): walker shows no step trend and cheetah has no clear gain. **(c)** PointMaze calibrated and uncalibrated success rates overlap within standard error (60 episodes per arm). **(d)** Model-versus-random signal in standard-error units is a diagnostic of the plain model-policy arms, not a DARC-versus-plain effect.

L THE STRUCTURE OF WHAT IS CALIBRATABLE

This appendix tests which correction is worth amortizing. The main-text Figure 5(a) shows the asymmetry: spread rescaling removes up to 33.2% of energy distance where the model is under-dispersed, whereas the tested mean correctors remove at most 2.6%. In panel (b), one global constant recovers almost all of the per-context optimum because the quantity corrected varies little across the tested contexts. Panels (c) and (d) then compare that cheap calibration against spending more sampler steps, including measured latency on a pixel-generating video model. The gate declines cartpole and pendulum on held-out anchors because their fits do not clear its error bar; it does not need their task identities.

M ADDITIONAL PRACTICAL CONTROLS

Against an implemented scheduler, and against generic corrections. At matched compute the calibrated sampler at four steps is 18.9 to 25.1% better in energy distance than a learned per-frame confidence router at its own 4.3-step operating point. The correction is specific to the task it was fitted on: a task's own constant removes 44.6% of the energy distance and at least 24.3% in all 15 cells, while another task's constant gives -30.4% and a random direction -32.2%. And the ceiling of §3.2 survives a change of readout: through energy distance, direct observation recovers 52.7% while no scalar exceeds 7.0% (Figure 12(b)).

Visual fidelity. The deployed DARC map is applied after the value head, so it leaves generated frames, their sharpness, and their PSNR exactly unchanged for a fixed sampler seed. The visual comparison therefore concerns solver steps alone. More steps reduce the simulator-image statistic gap from 202 to 151 on locomotion and from 311 to 122 on Jaco, while PSNR to realized futures falls from 25.70 to 24.64 dB and from 24.43 to 23.01 dB. These image endpoints have different targets; neither establishes a control benefit. Figure 16 reports them alongside the separate readout endpoints.

N FIDELITY AXES IN THE STORED ROLLOUTS

The stored records support three comparisons with different reference objects. *Numerical fidelity* compares the finite-step sampler $Q_{\theta,S}$ with its converged learned law $Q_{\theta,\infty}$ while initial noise, context, and action are fixed. *Environmental fidelity* compares a model law with the reset-and-replay reference P_{ref} under the specified actuator-noise process. *Readout fidelity* first applies the frozen value head and planner readout, then compares $F_r(V_{\#}Q_{\theta,S})$ with $F_r(V_{\#}P_{\mathrm{ref}})$. These axes coincide only under additional assumptions about the learned law and the readout. DARC does not make those assumptions, so every reported metric names both the law being compared and the transformation through which it is observed.

Figure 16 places four stored endpoints on these axes. The simulator-image statistic gap falls as solver steps increase, indicating movement toward the simulator's aggregate image statistics. PSNR to the realized future falls at the same time, showing that the sharper sample is not a more accurate prediction of the particular future that occurred. The normalized planner readout error also rises with the integration budget. The first two panels are image-space diagnostics; the third is the quantity consumed by planning. Their opposing trends are the measured reason that numerical convergence and visual realism cannot stand in for readout fidelity.

The final panel isolates the intervention from the renderer. For a fixed context, action, initial noise, and sampler budget, DARC reuses exactly the plain model frames and changes only the dispersion of value samples after V. Its image-statistic gap and PSNR are therefore identical to the one-step plain arm by construction, whereas its readout may change for nonlinear F_r. This distinction prevents an incorrect visual claim: the calibration improves the tested readout estimate; it neither sharpens frames nor repairs their pixel error.

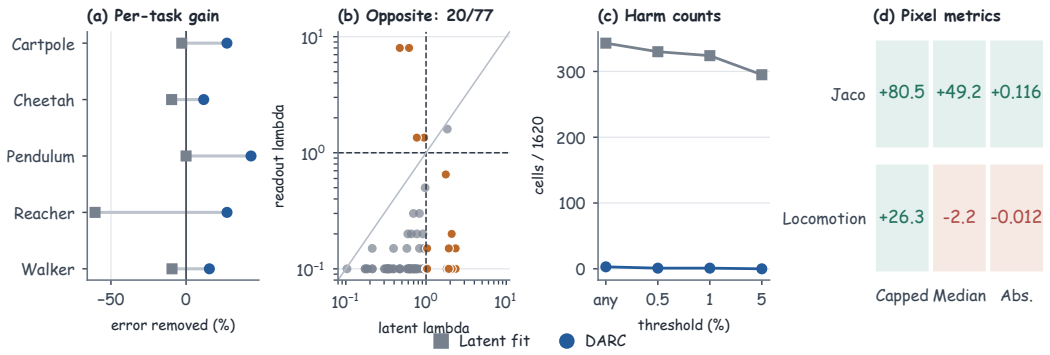


Figure 17: **Robustness of the readout-space comparison.** (a) Task-level mean relative error removed on the 135-model latent sweep (positive is better). (b) Latent- versus readout-optimal scalar in the 77 cells accepted by the original gate; orange points cross $\lambda = 1$, with the diagonal marking equal scalars. (c) Cells harmed above four relative-error thresholds, of 1620 paired cells. (d) DARC-minus-latent difference on the one-step pixel-model scene–readout aggregates: capped relative mean and uncapped relative median in percentage points, normalized absolute median in dimensionless units. Green means DARC is larger; the sign reversals in locomotion are retained.

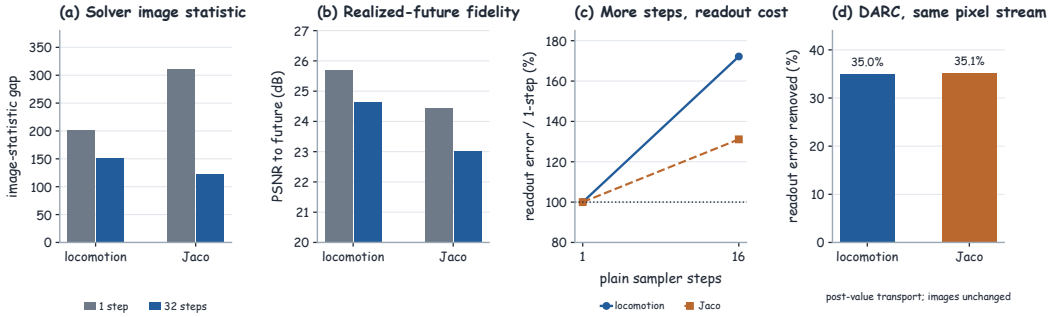


Figure 16: **Numerical, environmental and decision fidelity need not move together.** Four endpoints from the stored locomotion and Jaco evaluations: **(a)** simulator-image statistic gap (lower is better) and **(b)** PSNR to realized futures (higher is better) compare one with thirty-two solver steps. **(c)** Plain readout error at one and sixteen steps, normalized to the one-step error, shows the decision-statistic cost of additional integration. **(d)** The one-step DARC readout gain on each family is measured on the *same generated frames* as the one-step plain sampler. DARC changes value samples, not pixels; the image measurements in (a,b) therefore do not include a DARC image bar.

O STATISTICAL AND METRIC ROBUSTNESS

The 135 latent models comprise five tasks, three independent training seeds, and nine data/width/training-step configurations per seed. The 1620 step–readout cells share those models. A hierarchical bootstrap resamples task, seed, and configuration, retaining the twelve cells of each selected model. It gives a 95% interval of [15.8, 35.0] percentage points for mean readout error removed and [-40.1, -2.5] for latent fitting. The paired difference is 41.45 points [24.52, 66.05]; all five task aggregates and all 15 task–seed aggregates favor readout fitting. The seed-cluster screening check uses only three seed clusters per task and should be interpreted as a conservative sensitivity analysis, not a high-resolution estimate of a task-specific confidence bound.

Table 15 repeats the paired harm count at four thresholds. The large readout-versus-latent difference persists when harm requires more than a 1% increase in error.

Table 15: **Harm-threshold sensitivity on the 1620 latent-model cells.** Counts use paired corrected versus plain readout error.

Error increase defining harm	Readout fit	Latent fit
Any increase	3	343
More than 0.5%	1	330
More than 1%	1	324
More than 5%	0	295

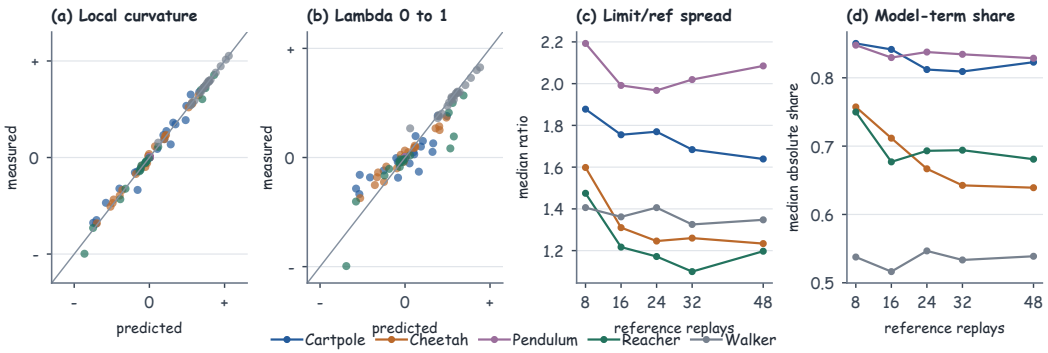


Figure 18: **Local curvature and replay-prefix checks on latent models.** (a) Exact autodifferentiated $\mathrm{tr}(H_V(m)\Sigma)$ versus the finite-difference second response centered at $\lambda = 0.1$ across 100 context–action pairs; both axes use the same signed $\mathrm{asinh}(x/0.05)$ display transform, so the diagonal denotes equality. (b) The same local coefficient predicts the full $\lambda = 0$ to 1 value-head response less precisely. (c) Per-task median converged-model-to-reference value-spread ratio against nested replay count; (d) per-task median absolute share of the learned-model log-dispersion term. Panels (c,d) use 250 candidate cells and up to 48 simulator replays; the pixel-model attribution has a distinct $R = 8$ reference and is not assigned a convergence claim by these panels.

The pixel results use a floor of -200% for the relative loss on each scene–readout aggregate. At one sampler step, the capped mean gives DARC minus latent differences of +80.5 points on Jaco and +26.3 points on locomotion. The uncapped medians give +49.2 and -2.2 points, respectively; median absolute-error differences normalized by each readout's median plain error give +0.116 and -0.012. The locomotion ordering therefore depends on the aggregate and does not hold under every robust metric.

The local curvature coefficient and the observed response have Pearson correlation 0.996 and 98% sign agreement near the expansion point. Extending the quadratic prediction from $\lambda = 0$ to 1 lowers correlation to 0.926 and sign agreement to 95%, consistent with the local scope of Proposition 3. In the replay-prefix check, the finite-inference term keeps its sign in all 250 candidates from $R = 8$ to 48, while the learned-model term keeps its sign in 84.8%. The median absolute change of ρ_∞ from $R = 32$ to 48 remains 5.4% (90th percentile 13.4%); the median absolute attribution-share change is 1.6 points. Neither result establishes complete convergence of the absolute ratio.

Noise-process perturbation. Figure 19 adds a separate latent-model check on walker and cartpole. Each of eight fresh contexts supplies five candidate actions. The same simulator state, action, and standard-normal perturbations are reused at 0.5, 1, and 2 times the original actuator-noise scale, with 24 replay futures per candidate. Model samples and their paired initial noise are fixed across noise scales. The learned-model median log-dispersion term moves from 0.652 to 0.258 on walker and from 0.987 to 0.364 on cartpole; the finite-step medians stay at -0.245 and -0.135. Candidate-wise opposing signs remain common, but the number of contexts with a top-two simulator-reference mean-value gap above two estimated standard errors falls at twice the noise. With only eight contexts per task, this is a sensitivity check, not a population claim or a re-estimate of pixel-video attribution.

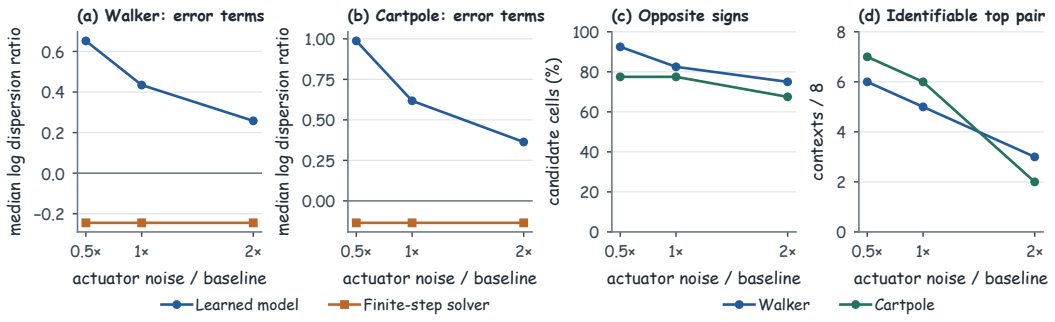


Figure 19: **Actuator-noise sensitivity under matched replay.** Eight held-out contexts per task, five candidate actions each, 24 simulator replays per action and noise scale. **(a,b)** Median learned-model and finite-solver log-dispersion terms on walker and cartpole. The solver term stays fixed because the model and sampled initial noise stay fixed. **(c)** Fraction of candidate cells with opposing term signs. **(d)** Contexts whose simulator-reference top-two mean-value gap exceeds twice its estimated standard error. This diagnostic does not measure DARC action-selection gain.

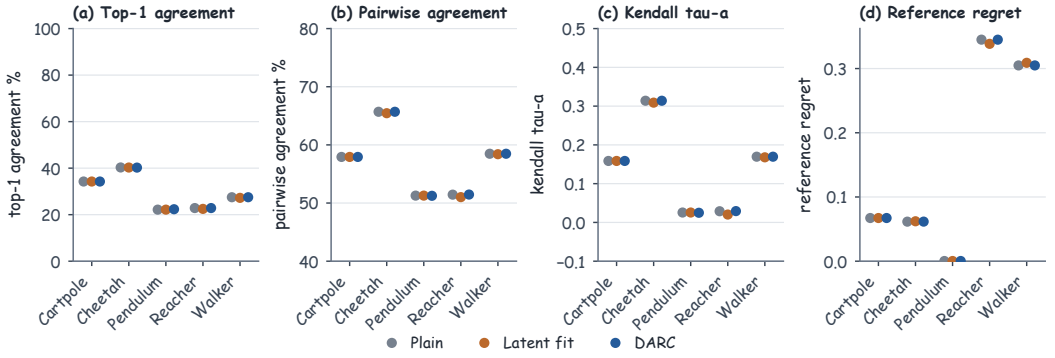


Figure 20: **Candidate-selection audit across tasks.** Four aligned panels report task-level action-selection diagnostics at the sampler budgets shown. All 135 latent models are evaluated on 125 held-out contexts, five candidate actions, and 24 independent simulator replays per candidate. Plain, latent-fit, and DARC curves overlap on the reported endpoints; Table 16 gives the selected numerical comparisons.

Table 16: **Selected candidate-selection results by sampler budget.** Top-1 and pairwise agreement are percentages; τ_a is Kendall's rank correlation with ties, and regret is in simulator-reference reward units (lower is better). The same held-out contexts and simulator replays are used for each method. The table supports no consistent action-selection gain.

S	Method	Top-1 (%)	Pairwise (%)	τ_a	Regret
1	Plain	29.06	56.43	.129	.16097
	Latent	28.09	56.02	.120	.16528
	DARC	29.10	56.38	.128	.16030
4	Plain	29.41	56.96	.139	.15580
	Latent	29.28	56.81	.136	.15543
	DARC	29.43	56.96	.139	.15580
32	Plain	28.39	56.41	.128	.16003
	Latent	28.34	56.34	.127	.16109
	DARC	28.25	56.32	.126	.16012

P ADDITIONAL RESULTS

Figures 21 and 22 repeat the matched comparison at the level of individual scenes and complement the pooled estimate in Figure 3. Its before/after panels show whether a gain is widespread or concentrated in a few large-error cells; its per-scene panels expose the cases in which a latent-variance fit improves or harms the readout. The step-count panels then test whether additional model evaluations provide the same correction, and the α sweeps reveal whether removing more of the candidate-relative offset helps throughout the tested range. It does not: pendulum worsens throughout the sweep, while finger and place brick have non-monotone responses. These failures are retained in the aggregate. The α experiment is a feature-space diagnostic; it does not test DARC's post-value sample transport. It limits any claim that removing the candidate-relative feature component always improves a decision. Figure 12(c,d) asks a separate resource question: how much offset energy differs across candidates, and what each kind of additional information costs. The measured wall-clock comparison keeps the decision-relevant signal distinct from the cost of a simulator reference.

Comparison across model families. The readout-space fit removes almost the same aggregate fraction of error on locomotion and Jaco (35.0% and 35.1%), despite their different scenes and value-head coverage. The latent-space fit does not transfer in that way: its aggregate gain is 19.7% on the six usable locomotion scenes and -10.2% on all four Jaco scenes. This is a matched-target comparison, not evidence that Jaco is universally harder: the model architectures and fitted readouts are held in their respective family, and the correction target is what changes within each family.

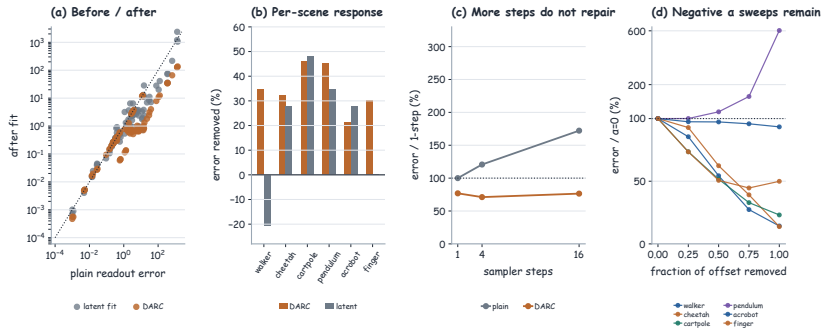


Figure 21: **Locomotion model, four matched tests in one row. (a)** Per-cell readout error before and after each correction, with the identity as the no-change reference. **(b)** Per-scene error removed in readout and latent fitting spaces. **(c)** Median readout error versus sampler steps for plain and corrected predictions. **(d)** The α sweep includes the pendulum exception. The aggregate readout fit removes 35.0% with 510/5184 cells harmed, versus 19.7% and 1336/5184 for latent fitting.

Adverse cells and limits. Readout fitting harms 510/5184 locomotion and 477/3456 Jaco cells; the smaller latent sweep's 1/1620 harm rate does not transfer to either pixel model. In the separate α intervention, pendulum reaches 604% of baseline error at $\alpha = 1$, while place brick reaches 296% at $\alpha = 0.25$ before recovering. These task-specific failures motivate held-out selection. The per-scene plots show where an aggregate gain conceals harm, and the candidate-choice and closed-loop audits report the downstream endpoints separately.

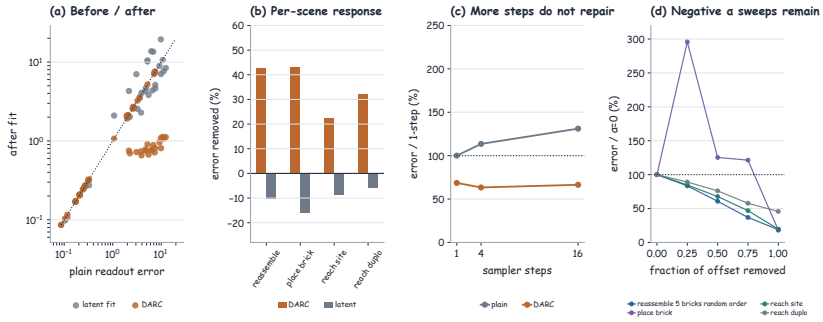


Figure 22: **Jaco manipulation model, the same four matched tests.** Panels (a–d) use the same definitions and ordering as Figure 21. Readout fitting removes 35.1% with 477/3456 cells harmed, versus −10.2% and 1129/3456 for latent fitting. The place-brick α curve initially rises before falling; this adverse response remains in the aggregate.

Scene-level comparison. Table 12 makes the family averages interpretable. On locomotion, readout fitting removes 34.6% of the walker error while latent fitting increases it by 20.5%; the corresponding gains are 32.4% and 27.7% on cheetah, 45.1% and 34.6% on pendulum, and 30.3% and 0.1% on finger. Cartpole and acrobot provide the counterexamples: the latent criterion removes 48.2% versus 46.0% on cartpole, and 27.8% versus 21.4% on acrobot. The pooled locomotion result therefore summarizes heterogeneous readout responses rather than a win in every scene. All four Jaco scenes favor readout fitting: its gains range from 22.3% on reach site to 43.1% on place brick, while the latent criterion raises error in every Jaco scene by 5.8% to 16.2%. Each scene in this table has 864 paired cells, so these comparisons have the same row denominator even though the scene-level error magnitudes differ.

Harm counts and interventions. The paired counts expose a second aspect of the comparison. On walker, readout fitting harms 148 of 864 cells and latent fitting harms 269; on place brick the counts are 112 and 362. Acrobot has the smallest readout harm count, 15, despite its lower mean gain than latent fitting. The Jaco readout gains therefore do not remove the need to inspect adverse cells. Figure 22(d) varies the candidate-relative offset removal coefficient α while keeping the sampled futures fixed. Place brick initially worsens before improving as more of that offset is removed. Figure 21(d) shows an even stronger adverse trajectory on pendulum. These sweeps evaluate a feature-space intervention; DARC's deployed value-sample transport is fitted separately in readout space. Their role is to test whether an identified error component supplies a universally safe correction direction.

Scope of the pixel-model evidence. The per-scene table uses readout-error reduction after the value head and reports all harmed cells under the stated threshold. The three low-R^2 locomotion tasks are visible in Table 11 and are excluded from the family aggregate; their diagnostic gains cannot be treated as evidence of reliable planning scores. The fitted scalar is chosen on disjoint contexts, and a different scalar is allowed for each task, sampler budget, and readout. Figures 21 and 22 establish a measured improvement in readout estimation under that protocol. Table 16 and Table 10 report the separate candidate-choice and closed-loop audits, whose null results limit the downstream claim.