EDGE: RESIDUAL-CONDITIONED CORRECTION ACQUISITION FOR VLA POST-TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

A failed robot trajectory is not automatically a useful training example. Under a fixed correction budget, the relevant question is which logged states deserve new expert actions. We study this offline acquisition problem for vision-language-action policies and introduce EDGE (Experience Discovery via Gap-aware Evaluation). A nominally trained, frozen latent world-action ensemble supplies three ranking signals: action-consequence surprise G, a teachability score T that measures the local mass of progress-improving action alternatives, and a confidence weight C that discounts uncertain residual transport. The observed transition residual anchors local counterfactuals without refitting the nominal models to the target pool. With the rollout pool, query count, replay mixture, and policy-update schedule held fixed, EDGE reaches $58.9 \pm 0.7\%$ on LIBERO-PRO and $56.2 \pm 0.7\%$ on PhysicalShift. A G-bandpass baseline whose center is tuned on development tasks and then frozen remains about $3\,\mathrm{pp}$ behind on ten separate validation seeds. More importantly, a separate same-snapshot PhysicalShift rollback study holds the nominal world-action model, progress verifier, 16 candidate actions, G, and budget fixed while changing only residual transport. Constant anchoring reduces the error of the estimated T and raises downstream success by $3.5\,\mathrm{pp}$; kernel decay further improves ranking and adds $0.7\,\mathrm{pp}$, with improvements under friction, mass, damping, and actuation-gain shifts but none under action delay. This factor-wise pattern connects the local transport assumption to where it succeeds and fails. With five mechanism seeds, statistical inference is limited, but the evidence links counterfactual prediction, acquisition ranking, and equal-budget post-training.

1 INTRODUCTION

Vision-language-action (VLA) models connect language-conditioned perception to robot actions (Zitkovich et al., 2023; Ghosh et al., 2024; O'Neill et al., 2024; Black et al., 2025); OpenVLA and its optimized fine-tuning recipe demonstrate the practicality of adapting such policies to manipulation tasks (Kim et al., 2025b;a), while SmolVLA explores a smaller and more accessible design (Shukor et al., 2025). Strong nominal behavior does not make every deployment failure equally informative. One failure may repeat an already represented behavior; another may require a long recovery sequence that a single corrective chunk cannot teach. Spending an equal expert budget on these states can yield very different post-training value.

We isolate *acquisition*: after a frozen policy has produced a shared deployment pool, which states should receive newly purchased corrections? This differs from changing the policy loss, collecting a larger pool, or deciding when to intervene online. A selector may inspect the logged observation, action, observed continuation, instruction, and episode outcome. It cannot inspect an expert correction before selecting its state. Each selected state costs one query, including states sampled from successful episodes.

Our acquisition rule is motivated by a *support-edge principle* (Fig. 1): useful corrections concentrate where the observed action consequence is unusual relative to nominal experience, but nearby actions still have recoverable structure. We operationalize these two requirements separately. Surprise G identifies departures from nominal consequences. A residual-conditioned progress comparison estimates a soft teachability score T. A confidence weight C discounts model spread and costly

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.

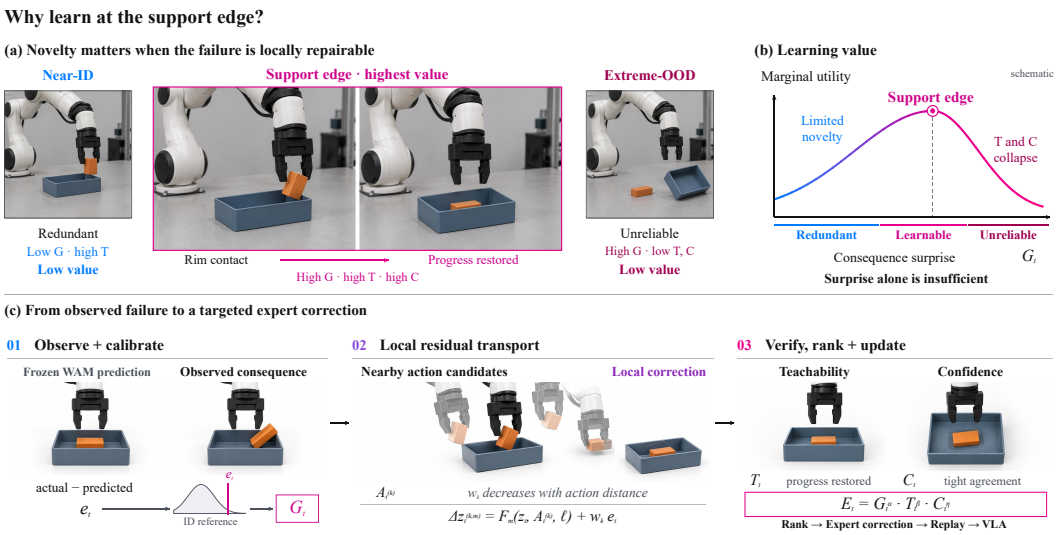

Figure 1: **EDGE selects support-edge failures: novel enough to matter, local enough to repair.** (a) Near-ID failures are redundant and extreme-OOD failures are unreliable; a support-edge failure combines surprise with local repairability. (b) Schematic: marginal learning value rises with action-consequence surprise only while local teachability and confidence hold, so surprise alone is insufficient. (c) The EDGE pipeline: calibrate the observed residual e_t into surprise G_t against nominal residuals, transport e_t to nearby candidate actions with distance-decaying weights w_k (ρ_k in Eq. (3)), then score local teachability T_t and confidence C_t, and query experts for the top-B states. The general score is $E_t = G_t^{\alpha} T_t^{\beta} C_t^{\eta}$; all reported experiments use unit exponents, $E_t = G_t T_t C_t$. Fig. 2 gives the controlled retraining evidence.

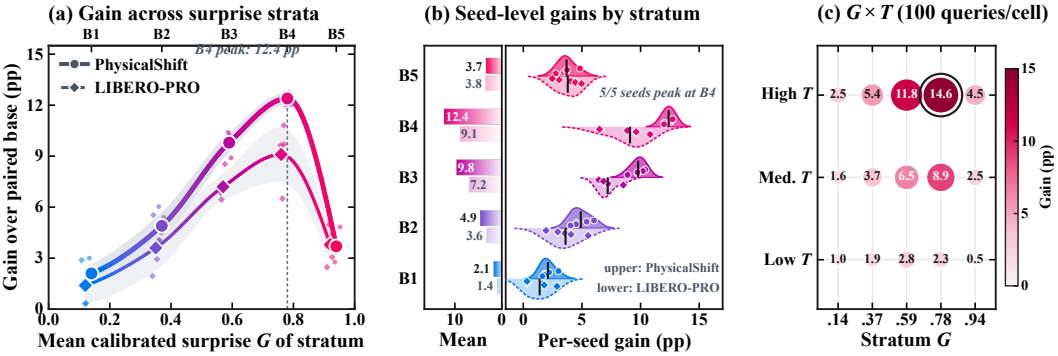

Figure 2: **Which correction strata are useful?** Failures are split into five equal-size G strata, B1 (lowest G) to B5 (highest); each stratum purchases 500 corrections from the plotted benchmark and updates a separate policy from the shared base checkpoint. Color encodes the stratum's mean G. (a) Mean paired gain against the stratum's mean G; ribbons are sample standard deviations over five seeds, small dots are individual seeds, and monotone interpolating lines guide the eye. (b) Mean gain per stratum (bars) and the five per-seed gains (split violins: PhysicalShift above, LIBERO-PRO below; ticks mark means). Every seed peaks in B4 on both benchmarks. (c) The complementary support/teachability grid with 100 corrections per cell; its absolute gains are not comparable with (a). The benchmarks have different bin populations and mean G coordinates.

transport of the observed residual to distant actions. Their product ranks states; the policy objective is unchanged.

Our contribution is a formulation and empirical study of corrective-label acquisition under matched supervision and policy-update budgets. We identify a support-edge regime in which corrections are most valuable, operationalize it through residual-conditioned experience valuation, and directly test when the residual-transport mechanism succeeds and fails. A same-snapshot PhysicalShift

rollback experiment performs this test on shared simulator snapshots, and a separate analytic stress test serves as a positive/negative control that delineates edge cases. The formulation keeps acquisition information, expert-query accounting, policy optimization, and statistical units explicit, so that gains can be traced to which states are corrected rather than to extra labels or a different policy update.

2 Related work and scope

DAgger studies learning on learner-induced states (Ross et al., 2011). Gated variants decide when control passes to an expert using human judgment (Kelly et al., 2019), a predicted deviation from the expert (Zhang & Cho, 2017), or ensemble disagreement (Menda et al., 2019); deployment systems reuse human interventions as a training signal (Mandlekar et al., 2020; Liu et al., 2023b), and calibrated uncertainty can decide when a robot should ask for help (Ren et al., 2023). ThriftyDAgger budgets interactive supervision using novelty and risk (Hoque et al., 2022); core-set acquisition emphasizes coverage (Sener & Savarese, 2018), and pool-based active learning more broadly selects labels by uncertainty, disagreement, or diversity (Settles, 2009; Houlsby et al., 2011; Gal et al., 2017; Ash et al., 2020). These works motivate several of our baselines, although their original interaction protocols differ from buying labels from a fixed, retrospectively logged pool; our risk–novelty and core-set rows adapt them to this setting (Appendix E).

Data valuation estimates each training example's contribution through Shapley values, influence functions, or datamodels (Ghorbani & Zou, 2019; Koh & Liang, 2017; Ilyas et al., 2022). In robot imitation, DataMIL uses datamodels to select existing imitation-learning data by its downstream value (Dass et al., 2026), and studies of demonstration quality and dataset mixtures ask which offline data help most (Belkhale et al., 2023; Hejna et al., 2025). Our acquisition unit is instead a new corrective label at a deployment state, with the same update rule across selectors. Interactive reinforcement post-training such as RIPT-VLA changes how policies learn from sparse success rewards (Tan et al., 2025); WoVR uses learned dynamics to support policy optimization in imagined rollouts (Jiang et al., 2026). These methods change how a policy learns, whereas EDGE fixes policy optimization and changes which corrections the policy receives, so they are complementary rather than budget-matched baselines.

Learned dynamics models support planning and policy learning in imagined or simulated experience (Chua et al., 2018; Hafner et al., 2020; Yang et al., 2024), and prediction error, ensemble disagreement, or confidence scores are classic novelty and out-of-distribution signals (Pathak et al., 2017; 2019; Burda et al., 2019; Hendrycks & Gimpel, 2017). Residual learning on top of a nominal controller or physics model is likewise common (Johannink et al., 2019; Zeng et al., 2020); EDGE instead observes a residual once and transports it locally rather than learning a correction model. Action-conditioned world models also support execution-time checking. CheckVLA, for example, uses a frozen action-conditioned world model and a conformally calibrated risk threshold to decide when to intervene during long-horizon execution (Liu et al., 2026). By contrast, EDGE uses an already observed transition to value an offline query before post-training. The distinction is the decision being optimized, the information available at that decision, and the fixed supervision budget. We evaluate with LIBERO (Liu et al., 2023a), the specified non-task subset of LIBERO-PRO (Zhou et al., 2025), and our PhysicalShift dynamics-shift suite (Appendix F). Comparisons to the full LIBERO-PRO leaderboard would require matching its task and perturbation protocol.

3 Problem and information budget

A frozen base policy $\pi_\theta(o_t, \ell)$ proposes a native action chunk $A_t = (a_t, \ldots, a_{t+H-1})$. A rollout pool $\mathcal{R}$ is collected before acquisition or policy updating. A candidate record is

$$x_t = (o_t, \ell, A_t, o_{t+\Delta}, y_\tau), \qquad \Delta = 4,$$

where y_τ is the episode's terminal binary outcome. Candidate extraction is fixed independently of selection scores and shared across selectors. Up to sixteen admissible anchor transitions per failed episode form a shared reservoir that every failure selector ranks.

Retrospective information. The selector uses $o_{t+\Delta}$ after the base action has executed. It is not available for a real-time decision at t. After selection, the correction oracle restores the snapshot at t

and returns a replacement chunk $A_t^\star$. The observed continuation is therefore permitted acquisition information; it is neither an expert label nor an online prediction.

Every non-Base method purchases exactly $B = 2{,}000$ native chunks from the joint LIBERO-PRO and PhysicalShift failure pool, and the resulting policy is evaluated on all three benchmarks. Failure-only samples failed episodes' candidate states; Success-only samples successful episodes' states; Random samples the shared full buffer. All score-based methods use the failed subset. A common within-trajectory exclusion of $|t - t'| \leq 8$ prevents adjacent selections. Post-training uses the same starting checkpoint, objective, update count, and 20% nominal replay fraction within each backbone. Equal chunk counts define a within-backbone budget; chunks with different native horizons need not cost the same human time.

4 Residual-conditioned experience valuation

4.1 Frozen nominal dynamics and finite-sample surprise

The frozen VLA encoder and a fixed nominal projection produce a $d = 512$ latent. Let $\tilde{z} = (z - \mu) \oslash (s + \epsilon)$, using nominal-training whitening statistics. An ensemble of $M = 3$ latent world-action models (WAMs) (Lakshminarayanan et al., 2017; Chua et al., 2018) predicts the latent change over the observed horizon,

$$F_m(\tilde{z}_t, A_{t:t+\Delta-1}, \phi(\ell)) \approx \tilde{z}_{t+\Delta} - \tilde{z}_t.$$

Only the executed Δ-step prefix is an input to this prediction: an unexecuted suffix must not alter the predicted Δ-step consequence. The ensemble is fitted on nominal transitions with Huber loss and bootstrap resampling. Neither the target pool nor later expert corrections refit the ensemble during a round.

With $\bar{F}_t = M^{-1} \sum_m F_m$, define

$$e_t = \tilde{z}_{t+\Delta} - \tilde{z}_t - \bar{F}_t, \qquad r_t = \|e_t\|_2 / \sqrt{d}, \tag{1}$$

and, following split-conformal ranking (Vovk et al., 2005; Angelopoulos & Bates, 2023), for n held-out nominal residual norms,

$$p_t = \frac{1 + \sum_{j=1}^{n} \mathbb{I}[r_j^{\mathrm{cal}} \geq r_t]}{n+1}, \qquad G_t = 1 - p_t = \frac{\sum_{j=1}^{n} \mathbb{I}[r_j^{\mathrm{cal}} < r_t]}{n+1}. \tag{2}$$

The strict inequality in the last expression specifies ties exactly. G lies on the lattice $\{0, 1/(n+1), \ldots, n/(n+1)\}$; values above the largest calibration residual all receive the same G, and we never add jitter to these ranks. Because G measures how far an observed consequence lies outside nominal experience, we also call it the support gap (hence Support-only and the support strata below). It is a marginal rank statistic, not a calibrated probability or a conformal guarantee under shift.

4.2 Local counterfactuals anchored by an observation

For $K = 16$ candidate chunks, including the executed one, let D_A be the squared diagonal Mahalanobis distance on the observed action prefix, using fixed nominal action variances. It is a sum, not a dimension-normalized average. Define

$$\rho_k = \exp\left[-\frac{D_A(A_t^{(k)}, A_t)}{2\sigma_A^2}\right], \qquad \hat{z}_{t+\Delta}^{(k,m)} = \tilde{z}_t + F_m(\tilde{z}_t, A_t^{(k)}, \phi(\ell)) + b_r \rho_k e_t, \tag{3}$$

where the action argument again denotes the observed prefix, $\sigma_A = 1$, and the default residual switch is $b_r = 1$. The residual supplies an observed local offset, while the kernel limits how far this offset is transported. It does not learn the changed action-response Jacobian. At the executed action, the *ensemble mean* exactly equals the observed next latent when $b_r = 1$; individual ensemble members need not equal it.

A nominally trained progress verifier $V(\tilde{z}, \phi(\ell)) \in [0, 1]$ uses temporal ranking pairs from successful demonstrations and binary terminal anchors from a separate nominal rollout set. Let $\bar{z}^{(k)}$ be the

ensemble-mean counterfactual latent and $v_t^+ = V(\tilde{z}_{t+\Delta}, \phi(\ell))$. Teachability is

$$q_k = V(\bar{z}^{(k)}, \phi(\ell)) - v_t^+, \qquad T_t = \frac{1}{K}\sum_{k=1}^{K} \sigma\left(\frac{q_k - \delta}{\tau}\right), \quad \delta = \tau = 0.05. \tag{4}$$

The thresholds are in the verifier's score units. T is a soft mass of locally improving alternatives, not the probability that an expert will recover the task. The executed candidate has $q = 0$ and contributes $\sigma(-1)/K$ rather than zero. Because T averages over sampled candidates, the candidate sampler is part of its definition.

4.3 Transport confidence

Define ensemble spread and residual-transport cost as

$$U_{\text{ens}} = \frac{1}{KMd}\sum_{k,m}\left\|\hat{z}_{t+\Delta}^{(k,m)} - \bar{z}^{(k)}\right\|_2^2, \tag{5}$$

$$U_{\text{tr}} = \frac{1}{Kd}\sum_{k}(1-\rho_k)^2\left\|e_t\right\|_2^2, \qquad C_t = \exp[-\gamma(U_{\text{ens}} + b_c U_{\text{tr}})]. \tag{6}$$

The default penalty switch is $b_c = 1$. The two switches b_r, b_c are independent, so the residual/penalty 2×2 ablation can turn each term off separately. $\gamma = \log 2/Q_{0.9}$, where $Q_{0.9} > 0$ is the nominal calibration quantile of $U_{\text{ens}} + U_{\text{tr}}$, is fixed for all four cells.

A shared additive residual cancels from ensemble dispersion. Consequently, U_{ens} cannot detect shared nominal-model bias, and U_{tr} is only a distance-dependent proxy for one component of transport error. Appendix B proves a local bound and identifies the smoothness and noise terms that C omits; C is therefore a heuristic weight rather than a certificate.

4.4 Selection and controlled updating

The default experience score is $E_t = G_t T_t C_t$. Candidates are processed by descending score, with lexical state-key tie-breaking for deterministic replay; a candidate is accepted only if no accepted state in its trajectory lies within eight control steps. Acquisition fails explicitly if the pool cannot supply B admissible states. Each accepted state receives one oracle query, with no post-query rejection of difficult examples. The policy is then updated using its native objective and the shared replay schedule. Appendix C specifies the algorithm; Appendix E distinguishes random tie-breaking for Support-only from deterministic replay ties.

5 Experimental protocol

Tasks and pools. Each of five seeds $\{11, 23, 47, 71, 89\}$ defines a separate acquisition pool and a set of evaluation resets shared by all selectors; every selector starts from the same base checkpoint. A pool has 3,200 LIBERO-PRO episodes (40 tasks $\times$ 4 non-task perturbation families $\times$ 20 resets) and 3,000 PhysicalShift episodes (10 tasks $\times$ 5 factors $\times$ 3 severities $\times$ 20 resets). LIBERO-ID uses the four standard suites. PhysicalShift independently changes delay, actuation gain, friction, mass, or damping in LIBERO's robosuite/MuJoCo simulation (Zhu et al., 2020; Todorov et al., 2012), the kind of physical parameters that dynamics randomization varies for sim-to-real transfer (Peng et al., 2018); specifications and aggregation are in Appendix F. No selector observes task-condition metadata for scoring; these labels are used only to construct the benchmark and the explicit quota diagnostic.

Optimization and supervision. OpenVLA-OFT is primary; SmolVLA is a backbone check. Within each backbone every non-Base selector uses 5,000 policy-update steps, global batch size 64, 80% queried corrections and 20% nominal replay. The oracle sees privileged simulator state only after selection and returns the backbone's native action chunk. All selectors share this oracle, whose corrections need not be perfect.

Comparators. In addition to random and single-signal rules, a G-bandpass ranks states by $-|G - c|$ with a fixed, nominally chosen center $c = 0.68$. A generic Recoverability comparator ranks G times the hard fraction of candidate nominal predictions exceeding observed progress by δ, without residual

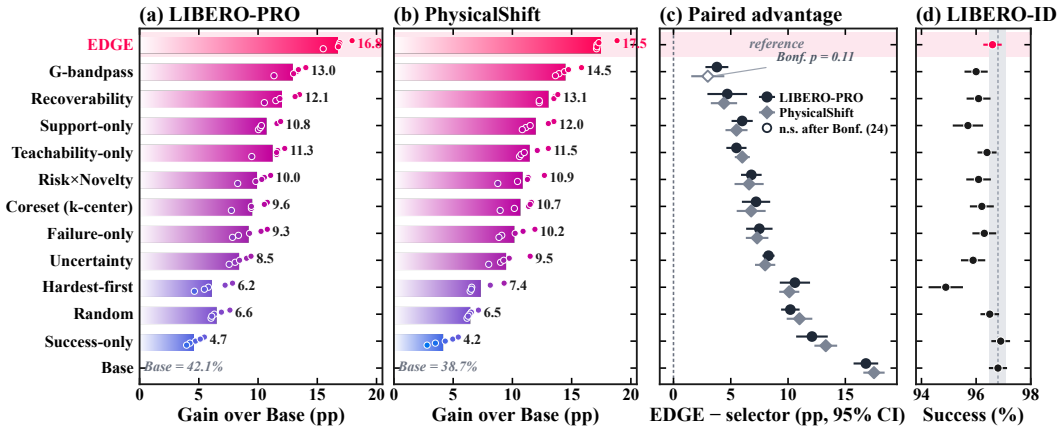

Figure 3: **Fixed-budget performance at 2,000 queries.** (a,b) Bars show the mean paired gain over Base and dots the five paired seeds, colored by gain. (c) Paired EDGE advantage with 95% t intervals ($\mathrm{df} = 4$). Filled markers remain below $p = 0.05$ after the conservative 24-comparison Bonferroni adjustment; the hollow marker (PhysicalShift vs. G-bandpass, $p = 0.108$) does not. (d) LIBERO-ID success (mean±SD); the band marks Base±SD. Selectors are sorted by mean shifted-benchmark success.

Table 1: Success (%, mean±sample standard deviation over five paired seeds). Every non-Base row purchases 2,000 chunks under the same policy-update schedule. LIBERO-PRO denotes the four-family subset specified in the text.

Selector	LIBERO-ID	LIBERO-PRO	PhysicalShift
Base	96.8 ± 0.3	42.1 ± 0.8	38.7 ± 0.7
Success-only	96.9 ± 0.3	46.8 ± 0.9	42.9 ± 0.8
Random	96.5 ± 0.3	48.7 ± 0.9	45.2 ± 0.8
Failure-only	96.3 ± 0.4	51.4 ± 1.0	48.9 ± 0.9
Hardest-first	94.9 ± 0.6	48.3 ± 1.1	46.1 ± 1.0
Uncertainty	95.9 ± 0.4	50.6 ± 0.9	48.2 ± 0.9
Support-only	95.7 ± 0.5	52.9 ± 0.9	50.7 ± 0.8
Teachability-only	96.4 ± 0.3	53.4 ± 0.8	50.2 ± 0.8
Risk×Novelty	96.1 ± 0.4	52.1 ± 0.9	49.6 ± 0.9
Coreset (k-center)	96.2 ± 0.4	51.7 ± 0.9	49.4 ± 0.9
Recoverability	96.1 ± 0.4	54.2 ± 0.8	51.8 ± 0.8
G-bandpass	96.0 ± 0.4	55.1 ± 0.8	53.2 ± 0.8
EDGE	96.6 ± 0.3	58.9 ± 0.7	56.2 ± 0.7

transport. Both are stronger tests than uniformly sampled failures, although they do not exhaust G-only or learned acquisition rules.

6 Results

6.1 Fixed-budget performance

Table 1 and Fig. 3 show $16.8/17.5$ pp gains over Base on LIBERO-PRO/PhysicalShift and $3.8/3.0$ pp over G-bandpass. ID success is essentially unchanged (96.6% versus 96.8% for Base).

Paired t intervals for the G-bandpass differences are $[2.79, 4.79]$ pp and $[1.55, 4.44]$ pp ($\mathrm{df} = 4$). Multiplicity changes the interpretation: across twelve non-EDGE comparisons on each of the two shifted benchmarks, a conservative 24-comparison Bonferroni adjustment yields $p = 0.011$ for LIBERO-PRO but $p = 0.108$ for PhysicalShift against G-bandpass. All five paired differences are positive in both cases, yet the two-sided exact sign-flip sensitivity has $p = 0.0625$. The point estimates are favorable, but a distribution-free or multiplicity-robust win on both benchmarks is not

Table 2: Seed-47 selector replay: composition of the selected states. All conditions select 2,000 states. Overlap divides the intersection by EDGE's budget; Jaccard uses the union. TVD is the total variation distance between the selected G distributions over the same eight bins.

Condition	B	Overlap / EDGE	Jaccard	G TVD	Trajectories
EDGE	2000	100.0%	100.0%	0.0%	1544
Permutation	2000	40.6%	25.5%	0.8%	1563
G-only mean	2000	55.6%	38.5%	32.5%	1560
Permutation + quotas	2000	42.2%	26.8%	5.7%	1560
G-only mean + quotas	2000	58.1%	40.9%	26.7%	1545

Table 3: Independent residual and transport-penalty switches. The same nominally calibrated γ is used throughout. Both-off is the joint removal ablation, not a pure residual-only intervention.

Residual	Penalty	LIBERO-PRO	PhysicalShift
off	off	55.0 ± 0.9	52.1 ± 0.8
on	off	57.7 ± 0.8	55.1 ± 0.7
off	on	55.9 ± 0.8	53.0 ± 0.8
on	on	58.9 ± 0.7	56.2 ± 0.7

established. Full comparisons, intervals, and assumptions appear in Appendix G. A G-bandpass whose center is tuned on disjoint development tasks and then frozen ($c = 0.75$) reaches 53.1% PhysicalShift success, against 56.2% for EDGE, on ten separate validation seeds (Appendix R).

6.2 The support-EDGE pattern and its limits

The PhysicalShift curve peaks in B4 (Fig. 2; exact values in Table 10): its gain is approximately $12.4\,\mathrm{pp}$, compared with $2.1\,\mathrm{pp}$ in B1 and $3.7\,\mathrm{pp}$ in B5. LIBERO-PRO shows an analogous non-monotonic pattern. These strata are formed using G, not the complete GTC score, avoiding a tautological ranking of E against itself. In the complementary 2-D grid with 100 queries per cell, gains rise with teachability in every G stratum and peak at high T and $G \approx 0.78$, while low-T cells stay below $3\,\mathrm{pp}$ at every G (Fig. 2c).

We read this pattern narrowly. Task, severity, and failure type can vary across support strata, so the equal-budget comparison estimates the utility of these *acquisition strata* rather than the effect of changing a state's G while keeping its other properties fixed. Longer optimization weakens, but does not remove, the peak-to-tail gap in the reported sweep (Appendix I).

6.3 Selection composition diagnostics

Table 2 asks how much of EDGE's selection G alone determines. We keep each state's G and either jointly permute (T, C) within value-based G bins or replace TC by its pool-level within-bin mean; the mean score is piecewise linear in G and uses no outcomes. Saturated ranks remain together, giving eight bins in this instance. Exact-quota variants reproduce EDGE's benchmark–task–factor–severity counts while retaining temporal exclusion.

Permutation and bin-mean selections overlap EDGE by 40.6% and 55.6%, with binned-G TVD of 0.8% and 32.5%: a nearly identical G distribution can coexist with substantial state replacement. Matching quotas reduces coarse composition differences but does not fix histories or within-cell G. The replay characterizes selection composition only (Appendix H).

6.4 Components, transfer, and query efficiency

Table 3 separates the kernel-residual construction from the transport penalty; Section 6.5 separates constant anchoring from kernel decay. Residual-on/penalty-off raises PhysicalShift success from approximately 52.1% to 55.1%; enabling the penalty as well gives 56.2%. The isolated penalty-on gain is smaller. All component removals and seed-level factorial contrasts are in Appendix J.

Table 4: Same-snapshot PhysicalShift rollback diagnostic over five separate paired seeds. Latent MSE ($\times 10^{-3}$) and soft-T MAE average candidate and state errors; overlap (%) compares the selected top 2,000 states with the rollback-derived reference ranking; regret is the mean reference-utility gap of the selected set.

Transport	latent MSE ($\times 10^{-3}$) ↓	soft-T MAE ↓	top-2k overlap (%) ↑	regret ↓
Nominal	10.565 ± 0.007	0.0674 ± 0.0008	80.7 ± 0.5	0.0135 ± 0.0005
Constant	3.903 ± 0.005	0.0495 ± 0.0005	85.5 ± 0.4	0.0079 ± 0.0005
Kernel	2.860 ± 0.002	0.0398 ± 0.0003	88.2 ± 0.4	0.0052 ± 0.0004

Table 5: Mechanism-only PhysicalShift post-training over five separate paired seeds. EDGE-N/C/K rank states by GT with Nominal, Constant, or Kernel transport; G, candidate sampling, budget, replay, update schedule, and $C = 1$ are shared.

Selector	PhysicalShift success	Δ vs. EDGE-C
EDGE-N	52.0 ± 0.6	-3.5
EDGE-C	55.5 ± 0.5	0.0
EDGE-K	56.2 ± 0.8	$+0.7$

On SmolVLA, EDGE reaches $51.8/49.7\%$, compared with $49.3/47.1\%$ for the strongest comparator, Recoverability. In leave-one-factor-out acquisition, the evaluated PhysicalShift factor is absent from the correction pool. The five-factor macro mean is 49.6% for EDGE, 47.0% for G-bandpass, and 48.3% for Recoverability; the margin narrows for unseen factors, and generalization beyond these shifts remains untested (Appendix K).

At all five measured query budgets, EDGE outperforms the two main comparators (Appendix L). Linear interpolation of the mean curves at an illustrative 53% PhysicalShift target gives approximately 1,111 queries for EDGE, 1,929 for G-bandpass, and 2,753 for Recoverability. The resulting ratios, 1.74 and 2.48, are interpolations between measured budgets rather than measured stopping times. Alternative correction channels and independent pools provide additional robustness comparisons (Appendix M).

6.5 Same-snapshot PhysicalShift residual-transport evaluation

We isolate the residual mechanism on the actual PhysicalShift acquisition interface. For every snapshot, Nominal, Constant, and Kernel share the same nominal WAM ensemble and progress verifier, the same executed action, and the same 16 candidate actions. We fix the confidence weight to $C = 1$, so the only change is whether the observed residual is ignored, copied to every candidate, or decayed by $\rho(a, a_t)$. Simulator rollback supplies the candidate continuation used to compute the reference soft-T^* with the same sigmoid average as Eq. (4). At the executed candidate, $D_A(a_t, a_t) = 0$ and $\rho = 1$, so Constant and Kernel coincide there. The reference difference q_k^* is the rollback verifier value minus the observed-reference value.

Table 4 shows that observation anchoring is the dominant step: Constant substantially improves on Nominal in all four diagnostics. Kernel then reduces soft-T error from 0.0495 to 0.0398, raises top-2k overlap from 85.5% to 88.2%, and cuts ranking regret by about one third relative to Constant. The effect is not uniform across dynamics (Fig. 4). Friction, mass, and damping show the clearest kernel-over-constant benefit, actuation gain a smaller one, and delay reverses slightly, consistent with a discontinuous response that violates the local smoothness premise.

The acquisition difference survives post-training when only the residual mode changes and each selector buys the same 2,000 corrections. Each PhysicalShift rate in Table 5 is reconstructed from 3,000 binary evaluation outcomes per seed (10 tasks × 5 factors × 3 severities × 20 resets); all seeds and modes use the same resets. Because this cohort has its own seeds and a PhysicalShift-only snapshot pool, its absolute rates are not directly comparable with Table 1. Constant anchoring accounts for roughly $3.5\,\mathrm{pp}$ over Nominal, while kernel decay adds a smaller $0.7\,\mathrm{pp}$ overall. Appendix P reports the factor-wise results. We therefore interpret the method as *observed residual conditioning*

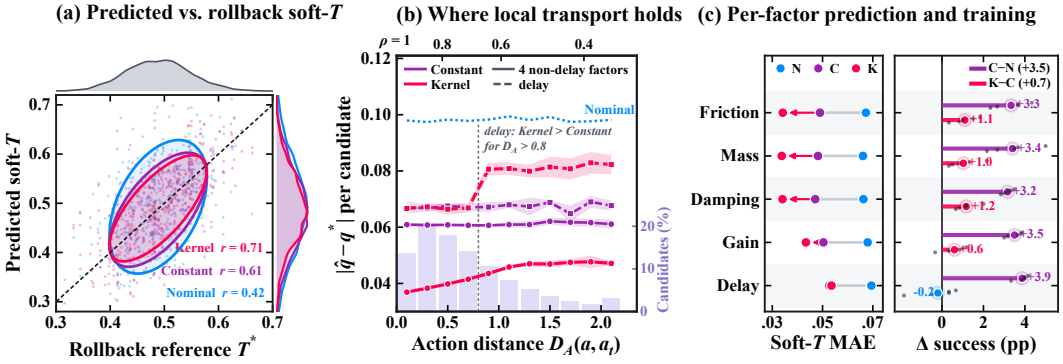

Figure 4: **Same-snapshot PhysicalShift residual transport, from prediction to post-training.** (a) Snapshot soft-T predicted by each transport mode against the rollback reference T^* over 20,000 shared snapshots; ellipses enclose 68% of snapshots under a Gaussian approximation, dots are a random subsample, margins show the distributions, and r is the Pearson correlation. (b) Candidate-level $|\hat{q} - q^*|$ against action distance for the four non-delay factors (solid) and delay (dashed); bands are 95% intervals across five seeds and bars show the candidate distance distribution. Kernel error grows with distance but exceeds Constant only under delay, where it jumps for $D_A > 0.8$. (c) Per factor: soft-T error of Nominal (N), Constant (C), and Kernel (K), and post-training success differences for observation anchoring (C−N) and kernel decay (K−C); small dots are seeds and legend values are overall differences. Kernel improves the smoother friction, mass, and damping factors, helps less under actuation gain, and does not help delay.

with a locality-dependent refinement rather than as a universally superior decay rule. The analytic stress test in Appendix Q provides controlled positive and negative checks.

7 Limitations and Conclusion

The scorer is ordinal and local. Its confidence weight cannot rule out shared bias, its visual latent may alias hidden physical state, and one observed residual cannot identify an arbitrary changed dynamics Jacobian. All reported tasks share one simulator family; real-robot and cross-simulator validation remain future work. Five paired seeds per experiment constrain statistical power, and the factor-wise pattern should not be read as a universal taxonomy of smooth versus discontinuous shift.

The practical findings are twofold. First, under a fixed correction budget, the complete acquisition rule outperforms the tested simple selectors, including a G-bandpass baseline with a development-tuned center (Appendix R). Second, same-snapshot rollback shows that most of the residual benefit comes from anchoring the nominal model to the observed transition, while local kernel decay is a smaller refinement that helps the smoother PhysicalShift factors and does not help delay. The analytic negative controls explain why this conditional behavior is expected rather than anomalous.

AI use statement

In this work, we used generative AI tools (large language models) for the following: (1) prose editing and grammar checking of draft text written by the authors; (2) feedback on experimental-design choices; (3) code review and artifact-integrity checking; (4) creation of conceptual task illustrations (Figure 1). Generative AI was *not* used to design the method, derive mathematical results, run experiments, or produce any quantitative data reported in this paper. All experimental results, statistical analyses, proofs, and final manuscript content were produced and verified by the authors. We take full responsibility for the correctness of all claims, text, code, and artifacts.

Ethics statement

EDGE ranks states for expert correction; it is not a safety monitor. Deploying a post-trained policy still requires independent monitoring and validation of the correction channel.

REPRODUCIBILITY STATEMENT

The supplement contains the reference implementation, the seed-level records behind every table, score-level selection replays, same-snapshot transport records, and scripts that regenerate and check all reported numbers. Raw camera streams and model checkpoints are not included because of their size. Appendix O summarizes what can be reproduced.

REFERENCES

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc Bellemare. Deep reinforcement learning at the edge of the statistical precipice. In *Advances in Neural Information Processing Systems*, volume 34, pp. 29304–29320, 2021.

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023.

Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *International Conference on Learning Representations*, 2020.

Suneel Belkhale, Yuchen Cui, and Dorsa Sadigh. Data quality in imitation learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 80375–80395, 2023.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, et al. π_0: A vision-language-action flow model for general robot control. In *Robotics: Science and Systems*, 2025.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *International Conference on Learning Representations*, 2019.

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*, volume 31, 2018.

Shivin Dass, Alaa Khaddaj, Logan Engstrom, Aleksander Madry, Andrew Ilyas, and Roberto Martín-Martín. DataMIL: Selecting data for robot imitation learning with datamodels. In *International Conference on Learning Representations*, 2026.

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, New York, 1993.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep Bayesian active learning with image data. In *International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1183–1192. PMLR, 2017.

Amirata Ghorbani and James Zou. Data Shapley: Equitable valuation of data for machine learning. In *International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2242–2251. PMLR, 2019.

Dibya Ghosh, Homer Rich Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, et al. Octo: An open-source generalist robot policy. In *Robotics: Science and Systems*, 2024.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020.

Joey Hejna, Chethan Anand Bhateja, Yichen Jiang, Karl Pertsch, and Dorsa Sadigh. ReMix: Optimizing data mixtures for large scale imitation learning. In *Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 145–164. PMLR, 2025.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *AAAI Conference on Artificial Intelligence*, volume 32, pp. 3207–3214, 2018.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2017.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2): 65–70, 1979.

Ryan Hoque, Ashwin Balakrishna, Ellen Novoseller, Albert Wilcox, Daniel S. Brown, and Ken Goldberg. ThriftyDAgger: Budget-aware novelty and risk gating for interactive imitation learning. In *Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, 2022.

Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Understanding predictions with data and data with predictions. In *International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9525–9587. PMLR, 2022.

Zhennan Jiang, Shangqing Zhou, Yutong Jiang, Zefang Huang, Mingjie Wei, Yuhui Chen, et al. WoVR: World models as reliable simulators for post-training VLA policies with RL. *arXiv preprint arXiv:2602.13977*, 2026.

Tobias Johannink, Shikhar Bahl, Ashvin Nair, Jianlan Luo, Avinash Kumar, Matthias Loskyll, et al. Residual reinforcement learning for robot control. In *International Conference on Robotics and Automation*, pp. 6023–6029, 2019.

Michael Kelly, Chelsea Sidrane, Katherine Driggs-Campbell, and Mykel J. Kochenderfer. HG-DAgger: Interactive imitation learning with human experts. In *International Conference on Robotics and Automation*, pp. 8077–8083, 2019.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. In *Robotics: Science and Systems*, 2025a.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, et al. OpenVLA: An open-source vision-language-action model. In *Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713. PMLR, 2025b.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1885–1894. PMLR, 2017.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems*, 2023a.

Huihan Liu, Soroush Nasiriany, Lance Zhang, Zhiyao Bao, and Yuke Zhu. Robot learning on the job: Human-in-the-loop autonomy and learning during deployment. In *Robotics: Science and Systems*, 2023b.

Yushan Liu, Peibo Sun, Xintao Chao, Zhenyang Yang, Yifan Xie, Lingfeng Zhang, et al. CheckVLA: Execution-time verification with action-conditioned world model for long-horizon mobile manipulation. *arXiv preprint arXiv:2607.26789*, 2026.

Ajay Mandlekar, Danfei Xu, Roberto Martín-Martín, Yuke Zhu, Li Fei-Fei, and Silvio Savarese. Human-in-the-loop imitation learning using remote teleoperation. *arXiv preprint arXiv:2012.06733*, 2020.

Kunal Menda, Katherine Driggs-Campbell, and Mykel J. Kochenderfer. EnsembleDAgger: A Bayesian approach to safe imitation learning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5041–5048, 2019.

Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, et al. Open X-Embodiment: Robotic learning datasets and RT-X models. In *International Conference on Robotics and Automation*, pp. 6892–6903, 2024.

Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 2778–2787. PMLR, 2017.

Deepak Pathak, Dhiraj Gandhi, and Abhinav Gupta. Self-supervised exploration via disagreement. In *International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5062–5071. PMLR, 2019.

Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In *International Conference on Robotics and Automation*, pp. 3803–3810, 2018.

Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, et al. Robots that ask for help: Uncertainty alignment for large language model planners. In *Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 661–682. PMLR, 2023.

Stéphane Ross, Geoffrey Gordon, and J. Andrew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635. PMLR, 2011.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018.

Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

Mustafa Shukor, Dana Aubakirova, Francesco Capuano, Pepijn Kooijmans, Steven Palma, Adil Zouitine, et al. SmolVLA: A vision-language-action model for affordable and efficient robotics. *arXiv preprint arXiv:2506.01844*, 2025.

Shuhan Tan, Kairan Dou, Yue Zhao, and Philipp Krähenbühl. Interactive post-training for vision-language-action models. *arXiv preprint arXiv:2505.17016*, 2025.

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*, pp. 5026–5033, 2012.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, New York, 2005.

Sherry Yang, Yilun Du, Seyed Kamyar Seyed Ghasemipour, Jonathan Tompson, Leslie Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *International Conference on Learning*

Representations, 2024.

Andy Zeng, Shuran Song, Johnny Lee, Alberto Rodriguez, and Thomas Funkhouser. TossingBot: Learning to throw arbitrary objects with residual physics. *IEEE Transactions on Robotics*, 36(4):1307–1319, 2020.

Jiakai Zhang and Kyunghyun Cho. Query-efficient imitation learning for end-to-end simulated driving. In *AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Xueyang Zhou, Yangming Xu, Guiyao Tie, Yongchao Chen, Guowen Zhang, Duanfeng Chu, Pan Zhou, and Lichao Sun. LIBERO-PRO: Towards robust and fair evaluation of vision-language-action models beyond memorization. *arXiv preprint arXiv:2510.03827*, 2025.

Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Kevin Lin, et al. robosuite: A modular simulation framework and benchmark for robot learning. *arXiv preprint arXiv:2009.12293*, 2020.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2165–2183. PMLR, 2023.

A NOTATION AND CODE OVERVIEW

Table 6: Notation used in the paper.

Symbol	Meaning
H, Δ	Native correction horizon; observed transition horizon ($\Delta = 4$)
M, K, d	Ensemble size 3; candidate count 16; latent dimension 512
e, r, G	Observed residual; normalized norm; fixed-calibration surprise
ρ, b_r, b_c	Distance kernel; independent prediction and penalty switches
q, T, C, E	Progress difference; soft teachability; confidence weight; product score
B	Expert action-chunk budget, 2,000 in the main comparison

In the supplement, `edge/core.py` computes G, T, C, and E from nominal model predictions and observed latent changes; `edge/selection.py` implements the greedy and exact-quota selectors on state scores and identifiers without reading evaluation outcomes; and `scripts/analyze_results.py` regenerates the numerical tables from the released seed-level records.

B LOCAL RESIDUAL-TRANSPORT ANALYSIS

Consider one fixed underlying state and instruction. Write the conditional latent response as $f_*(a)$ and the nominal ensemble mean as $\bar{F}(a)$. Let $b(a) = f_*(a) - \bar{F}(a)$ be their discrepancy, and let the observed residual at a_0 satisfy $e = b(a_0) + \xi$, where ξ includes observation noise and realized transition variation. The transported prediction is $\hat{f}(a) = \bar{F}(a) + \rho(a, a_0)e$.

Proposition 1 (Local discrepancy bound)**.** *Suppose b is L_b-Lipschitz in a standardized action metric $h(a, a_0)$ over the neighborhood considered, $0 \le \rho \le 1$, and $\|\xi\| \le \varepsilon$. Then*

$$\left\| f_*(a) - \hat{f}(a) \right\| \le L_b h(a, a_0) + (1 - \rho) \|e\| + \varepsilon. \tag{7}$$

If the progress function is L_V-Lipschitz on the relevant latent region, replacing a true candidate latent by its prediction changes the candidate-minus-observed progress difference by at most L_V times the right-hand side.

Proof. The exact identity is

$$f_*(a) - \hat{f}(a) = b(a) - b(a_0) + (1 - \rho)e - \xi.$$

Apply the triangle inequality, the Lipschitz assumption, and the noise bound. The progress statement follows by applying the Lipschitz inequality to the candidate term; the observed reference is the same in both differences. □

For two candidates, a predicted progress margin larger than the sum of their error bounds would preserve their ordering. This implication is conditional: EDGE does not estimate L_b, L_V, or a uniform noise bound. In particular, U_{tr} resembles only the squared residual-distance term in Eq. (7). It omits the unknown local change of discrepancy and stochastic variation. Averaging across candidates is not a worst-case bound.

A constant-discrepancy counterexample. If $b(a) \equiv b_0$ and $\xi = 0$, constant residual transport is exact, whereas kernel transport has error $(1 - \rho)b_0$. Thus Proposition 1 decomposes error; it cannot establish superiority of decay. Appendix Q includes this counterexample as a reproducible negative control for the kernel.

What anchoring does and does not guarantee. At $a = a_0$ and $\rho = 1$, the ensemble-mean prediction equals the observed continuation. The true conditional response may still differ by ξ. Adding the same ρe to every model leaves their variance unchanged. Three models can agree perfectly on the wrong response at another action, so $U_{\mathrm{ens}} = 0$ can coexist with a large counterfactual error. The unit tests exercise this case. Discontinuous actuation delay, contact changes, and hidden-state aliasing can invalidate local smoothness even when the kernel distance is small.

Why an interior utility peak is plausible, not necessary. Let R denote the event that a useful local correction exists. The law of total expectation gives

$$\mathbb{E}[\Delta V \mid G = g] = \mathbb{E}[\Delta V \mid G = g, R]P(R \mid g) + \mathbb{E}[\Delta V \mid G = g, \neg R](1 - P(R \mid g)).$$

Increasing novelty and declining local recoverability can produce a peak, but their product need not turn over: relative rates and the second term matter. For a differentiable positive product $u(g) = n(g)r(g)$, a stationary point requires $(\log n)' = -(\log r)'$, and it is a maximum only with the appropriate derivative sign change. We neither assume this crossing universally nor derive a unique optimal G from it. The empirical strata, not this decomposition, supply the evidence for a peak in the reported setting.

C Acquisition algorithms and replay

Algorithm 1 One offline, matched-budget EDGE round

Require: frozen base policy, fixed nominal WAM ensemble, progress verifier, and calibration, shared logged pool, budget B

1: extract the shared candidate reservoir without using selection scores
2: **for** each failed-state candidate **do**
3: compute observed e, r, G with conservative finite-sample ties
4: evaluate K local action prefixes, including the executed prefix
5: compute T, C, and $E = GTC$ with fixed nominal hyperparameters
6: **end for**
7: order candidates by descending E, then lexical state key
8: greedily accept a state only when no accepted same-trajectory state is within eight steps
9: fail explicitly unless exactly B distinct states can be accepted
10: query each accepted snapshot once; retain every returned correction
11: update the policy with its native objective, fixed nominal replay fraction, and fixed steps

The released instance contains 13,000 scored failure anchors, a 6,200-episode acquisition registry, and ordered records for three greedy selectors plus two quota diagnostics. Raw images, per-model candidate predictions, and oracle action chunks are not included, so T and C enter the replay as recorded scores; `edge/core.py` computes them from model outputs.

With $n = 20{,}000$ calibration residuals, the largest attainable G is $20{,}000/20{,}001$. Exactly 3,640 of the 13,000 released residuals exceed the calibration maximum, so value-based quantile boundaries collapse to eight bins. In these bins and in all selection scores, neither raw residual order nor arbitrary row order splits a saturated G tie; only the equal-size support strata (Appendix I) use raw residual order as a secondary key. Tie-breaking in greedy selection is a separate, explicit operation after the score has been defined.

D Model interface and post-training specification

Nominal WAM ensemble. The nominal latent interface is a 512-dimensional projection of the frozen, language-conditioned visual representation. Projection and whitening parameters are fixed from the nominal training side before the calibration partition is evaluated. Each ensemble member has four transformer blocks, width 512, eight heads, MLP ratio four, and dropout 0.1. An action-prefix token sequence is cross-attended with state/instruction features. The target is the observed four-step latent delta, fitted with Huber loss. The nominal schedule is 80 epochs, AdamW learning rate 3×10^{-4}, weight decay 10^{-4}, batch size 256, cosine decay, and gradient clipping at 1. Bootstrap resampling and initialization differ across the three members; calibration examples are excluded from fitting.

Progress verifier. A 512–512–256–1 MLP with GELU and dropout 0.1 supplies a bounded score. Successful demonstration pairs separated by at least four steps enter a Bradley–Terry ranking objective. Separate nominal rollout terminal states supply binary anchors. The ranking and anchor losses are

weighted 1 and 0.5. Temporal ordering is a surrogate: tasks with necessary reversals or long non-monotonic subgoals may violate it. No dense privileged reward or target outcome is used to fit this verifier.

Candidate actions. OpenVLA-OFT uses local perturbations around the executed chunk, with Gaussian scale $0.25s_j$ per continuous action dimension and clipping at action bounds. The executed candidate is always included. Discrete gripper decisions retain the backbone's own action representation. SmolVLA alternatives use its native sampler and are restricted to a local action neighborhood. Distances and verifier predictions use the same observed prefix. Candidate sampling affects T; it is not a policy-independent physical recoverability estimator.

OpenVLA-OFT correction update. Post-training starts from the public LIBERO task-union OpenVLA-OFT checkpoint (Kim et al., 2025a) named in the configuration. Only rank-32 LoRA adapters are updated with the continuous-action L1 objective. AdamW uses learning rate 5×10^{-5}, weight decay 10^{-4}, 250 warm-up steps, cosine decay to 5×10^{-6}, global batch size 64, clipping at 1, and 5,000 updates.

SmolVLA preparation and update. SmolVLA starts from `lerobot/smolvla_base`; a common nominal task-union policy is prepared before acquisition, with 100,000 updates, batch size 64, and a frozen vision-language backbone. The action expert and state projection are trainable. AdamW uses betas $(0.9, 0.95)$, learning rate 10^{-4}, weight decay 10^{-10}, 1,000 warm-up steps, and cosine decay to 2.5×10^{-6}. Correction post-training retains the native conditional flow-matching objective, freezes the same backbone, and uses 5,000 updates with 500 warm-up steps and clipping at 10.

Correction channel and replay mixture. The task-level oracle restores the selected snapshot and uses privileged poses, robot state, and unsatisfied goal predicates only after acquisition. Its primitive library includes reach/pregrasp, grasp, lift/transport, place/release, articulation, toggle/rotation, and collision retraction. It returns one native chunk. Failed or poor corrections still count and are not filtered after querying. Sampling with replacement draws 80% correction examples and 20% nominal examples in expectation, i.e., 12.8 replay examples per batch of 64 on average.

E BASELINE DEFINITIONS

Every score-based rule below ranks the same failed-state reservoir. Random, Failure-only, and Success-only uniformly randomize their eligible records before the shared temporal exclusion. Their selected states are all charged an expert query.

Table 7: Score or selection rule of each compared selector.

Selector	Score or selection rule
Hardest-first	Descending continuous residual norm r.
Support-only	Descending calibrated G, with seed-fixed uniform tie-breaking.
Uncertainty	Executed-action nominal ensemble variance.
Teachability-only	Residual-conditioned T.
Risk×Novelty	$G[1 - V(\tilde{z}_{t+\Delta}, \phi(\ell))]$.
Coreset	Greedy maximum minimum distance in normalized frozen embeddings, from a seed-fixed initial state.
G-bandpass	$-\lvert G - c\rvert$ with center $c = 0.68$, fixed from a separate nominal failure set.
Recoverability	$GK^{-1} \sum_k \mathbb{I}[V(\tilde{z}_t + \bar{F}(A^{(k)})) > v_t^+ + \delta]$, without transport.
EDGE	GTC.

Monotone-rank caveat. Without calibration ties, Support-only and Hardest-first have the same ordering. Their difference is the treatment of large tied groups, especially the saturated tail, not additional information in G. A gap between these rows therefore cannot establish that a monotone transformation of residuals discovers a better notion of difficulty.

The nominal bandpass center is one heuristic choice. A stronger comparison tunes the center on disjoint development tasks and freezes it before test; the supplement includes the complete development outcomes, frozen-center record, and validation results (Appendix R). Recoverability uses the same candidate count and progress surrogate but omits observed-residual anchoring. Our core-set and risk–novelty rules are controlled adaptations motivated by prior acquisition work (Sener & Savarese, 2018; Hoque et al., 2022).

F BENCHMARKS, COUNTS, AND SPLIT DEFINITIONS

LIBERO-ID macro-averages four ten-task suites with 50 reset trials per task, giving 2,000 outcomes per seed. OpenVLA-OFT follows its center-crop convention. The reported LIBERO-PRO subset uses 40 tasks and four non-task families (object, initial-state/position, instruction/semantic, environment), with 20 evaluation resets per task–family cell, giving 3,200 outcomes. Task-generalization and compounded perturbation modes are excluded from that main number.

PhysicalShift uses all ten LIBERO-10 tasks, five independent factors, three severities, and 20 resets per cell, giving 3,000 outcomes. No condition combines factors in the main suite. Nominal values not listed below are retained. The LIBERO-10 task set is not performance-filtered. It covers paired-object basket placement, stove/moka-pot manipulation, bowl/cabinet placement, two-mug placement, book/caddy placement, mug/pudding placement, and microwave placement. The exact task instructions and factor implementation specification are included in `configs/physicalshift_spec.yaml`.

Table 8: PhysicalShift single-factor specification.

Factor	Mild	Moderate	Severe
Action delay (control steps)	1	2	3
Actuation gain multiplier	0.90	0.80	0.70
Contact-friction multiplier	0.80	0.60	0.40
Movable-object mass multiplier	1.25	1.50	2.00
Robot joint-damping multiplier	1.25	1.50	2.00

Within a seed, reset identifiers are shared across selectors for paired evaluation, but acquisition and evaluation identifiers are disjoint. Equal numbers of resets in every cell make macro-averaging equivalent to pooling binary outcomes within the specified benchmark. The integer-count file therefore stores the exact k/N representation rather than introducing intermediate decimal rounding. `scripts/verify_artifact.py` checks this identifier separation.

G STATISTICAL PROTOCOL

The analysis unit is a paired experiment seed, not an individual reset or a table cell. For seed differences $d_1, \ldots, d_5$, the reported interval is

$$\bar{d} \pm t_{0.975,4}\, s_d/\sqrt{5}.$$

It relies on independent seed-level draws and approximate normality of their differences. Pool collection, scoring, correction sampling, and policy training are coupled within a seed. Few-seed evaluation is a known reliability risk in deep learning (Henderson et al., 2018; Agarwal et al., 2021). Five seeds cannot establish normality, and tighter episode-level intervals would not add independent policy-training replicates.

The supplementary CSV contains all 36 main comparisons across three benchmarks, unadjusted t intervals, Bonferroni and Holm (Holm, 1979) adjustments within each benchmark, and 24-way adjustments across the two shifted benchmarks. Choosing the empirically strongest comparator after seeing all rows cannot be treated as one prospectively isolated hypothesis. Under the less conservative twelve-comparison per-benchmark Bonferroni convention, PhysicalShift vs. G-bandpass has adjusted $p \approx 0.054$; under the 24-comparison family it is ≈ 0.108.

Table 9: Paired comparisons against the two main comparators. $24p_t$ is capped at one and treats all twelve non-EDGE comparisons, including Base, across both shifted benchmarks as one family.

Benchmark	Comparator	Δ (pp)	95% t CI	p_t	$24p_t$ capped	p_{flip}
LIBERO-PRO	G-bandpass	3.79	$[2.79, 4.79]$	0.0005	0.011	0.0625
LIBERO-PRO	Recoverability	4.69	$[2.96, 6.43]$	0.0017	0.040	0.0625
PhysicalShift	G-bandpass	2.99	$[1.55, 4.44]$	0.0045	0.108	0.0625
PhysicalShift	Recoverability	4.41	$[3.26, 5.55]$	0.0004	0.011	0.0625

Exact small-sample sensitivities. The analysis enumerates all 2^5 sign flips of the paired differences. Under the sign-symmetry null, the smallest possible two-sided probability is $2/32 = 0.0625$, even if every difference is positive. An exact enumeration of all 5^5 seed-bootstrap resamples (Efron & Tibshirani, 1993) is also included as a descriptive sensitivity; a five-point bootstrap is not a remedy for the limited number of independent runs. Primary conclusions emphasize effect estimates and uncertainty rather than a binary acceptance threshold.

Per-reset binary outcomes are released for EDGE seed 47; the 8,200 records reconstruct its three benchmark totals.

H Permutation and exact-quota diagnostics

Estimator and randomness. Pool quantile boundaries are computed without consulting any selected set. Repeated edges collapse and equal G values stay together. Within each resulting bin, keys are sorted lexically and NumPy's generator with seed 7041 jointly permutes the (T, C) pairs for the released pool. Thus file-row permutation cannot change the result. The bin-mean control assigns $E_i = G_i\overline{TC}_{b(i)}$. It uses a pool-level statistic of the recorded T, C values but no downstream outcomes. It is G-only conditional on the fitted bin means, not independent of the pool used to estimate those means.

The descriptive counts are in Table 2; the ordered state records and all cell allocations are included in the supplement.

What quota matching controls. For each benchmark–task–factor–severity cell, the quota equals EDGE's count. The diagnostic maximizes total surrogate score subject to those exact quotas and the same temporal exclusion. A cardinality-constrained weighted-interval dynamic program solves each trajectory; a second dynamic program combines trajectories within a cell. Unlike the main selector, this is an exact quota-constrained optimizer, not the same unconstrained greedy algorithm.

Quota cells include the task dimension, and all 2,000 selected states are accounted for. Quota feasibility is checked, and infeasibility is an error rather than a silently reduced budget. Since the quotas themselves are derived from EDGE, they transfer some information from EDGE into the control. They also leave within-cell G and trajectory-history differences unconstrained. The replay therefore controls coarse composition only.

Record format. Every recorded selection includes its intrinsic G, T, C, E, the selection score actually used, and its acceptance rank; a state-record digest covers the serialized score and key content.

I Support strata and optimization budget

The one-dimensional support comparison purchases 500 corrections per G stratum and starts each update from the same base checkpoint. Strata are ordered by G, with raw residual order used only as an explicitly defined secondary key for equal-size stratification when a tie must be partitioned. This secondary stratification key does not change the calibrated G itself or the value-based bins used in permutation diagnostics. Different analyses therefore need not have the same bin boundaries.

Table 10: Equal-query support strata. Each bin receives 500 new queries and a separate policy update from the same base checkpoint. Gain uncertainty is the standard deviation of paired gains. PRO and PS have different bin populations.

Bin	Mean G (PS)	PS success	PS gain (pp)	PRO success	PRO gain (pp)
B1	0.14	40.8 ± 1.2	2.1 ± 0.5	43.5 ± 1.0	1.4 ± 1.1
B2	0.37	43.6 ± 1.0	4.9 ± 0.8	45.7 ± 1.0	3.6 ± 1.3
B3	0.59	48.5 ± 1.1	9.8 ± 0.6	49.3 ± 1.1	7.2 ± 0.8
B4	0.78	51.1 ± 0.9	12.4 ± 0.3	51.2 ± 0.9	9.1 ± 1.6
B5	0.94	42.4 ± 1.3	3.7 ± 0.7	45.9 ± 1.2	3.8 ± 1.0

The support/teachability grid supplies 100 corrections per cell. Both analyses match optimizer steps within their budgets but do not hold the task-condition mixture fixed as G varies; a condition-matched design would additionally require equal feasible query counts in every stratum.

Table 11: Gain over the corresponding base at extended optimization. The 80k-step comparison has three paired seeds; the shorter schedules have five. All rows are fixed-step results.

Bin	Optimizer steps	n seeds	Gain over base (pp)
B4	5000	5	12.4 ± 0.3
B4	40000	5	13.8 ± 0.4
B4	80000	3	14.1 ± 0.6
B5	5000	5	3.7 ± 0.7
B5	40000	5	7.2 ± 0.7
B5	80000	3	8.4 ± 0.5

The peak-to-tail advantage attenuates with longer training, so the lower value of the extreme tail depends on the optimization budget; the tail is not shown to be unlearnable.

J Ablations and model adaptation

Table 12: Component ablations under the main query and policy-update budgets. "w/o residual transport" denotes both residual anchoring and transport penalty off; Table 3 is the appropriate isolated comparison.

Variant	LIBERO-PRO	PhysicalShift
EDGE	58.9 ± 0.7	56.2 ± 0.7
w/o support surprise G	54.3 ± 0.8	51.1 ± 0.9
w/o teachability T	53.6 ± 0.9	51.5 ± 0.8
w/o confidence C	56.5 ± 0.8	53.6 ± 0.9
w/o residual transport	55.0 ± 0.9	52.1 ± 0.8
Observation-only novelty	54.7 ± 1.0	50.8 ± 0.9
No action conditioning in WAM	53.8 ± 0.9	49.9 ± 1.0
Single WAM	56.2 ± 0.8	53.0 ± 0.9
No temporal suppression	57.1 ± 0.9	54.0 ± 0.8

The per-seed factorial file exposes residual-on/penalty-off, residual-off/penalty-on, both-off, and both-on. `factorial_paired_statistics.csv` reconstructs the two simple effects and their interaction without treating the four cells as independent observations.

Fitting residual dynamics to the deployment pool is a plausible alternative. It changes the training data, the optimization budget, and the definition of surprise relative to the frozen nominal construction, so this comparison measures the effect of pool fitting in our setting rather than a general preference for freezing.

Table 13: A separate model-adaptation comparison. Target-pool fitting changes the frozen-model constraint, so these rows are not used to define the primary protocol.

Model variant	LIBERO-PRO	PhysicalShift
EDGE (frozen WAM, kernel transport)	58.9 ± 0.7	56.2 ± 0.7
G frozen + pool-fitted residual T,C	57.6 ± 0.5	55.4 ± 0.9
All pool-fitted (G,T,C)	52.8 ± 0.6	51.3 ± 0.7

K BACKBONE AND UNSEEN-FACTOR TRANSFER

Table 14: SmolVLA results over five paired seeds for the selectors run on this backbone.

SmolVLA selector	LIBERO-PRO	PhysicalShift
Base	36.7 ± 1.1	33.8 ± 1.0
Failure-only	44.9 ± 1.0	42.6 ± 1.1
EDGE	51.8 ± 0.9	49.7 ± 0.9
Support-only	46.8 ± 1.0	44.5 ± 1.0
Teachability-only	47.3 ± 1.0	44.0 ± 1.1
G-bandpass	48.7 ± 0.9	46.5 ± 1.0
Recoverability	49.3 ± 0.9	47.1 ± 0.9

Backbone-specific policy objectives and native chunk horizons differ. The transfer question is whether the ranking advantage survives changing the backbone under a matched budget within that backbone, not whether one model is universally more query-efficient than the other.

K.1 LEAVE-ONE-FACTOR-OUT ACQUISITION

Table 15: Leave-one-factor-out (LOFO) success. The macro standard deviation is computed after averaging held-out factors within a seed, not by averaging their standard deviations.

Selector	Delay	Gain	Friction	Mass	Damping	Macro
Base	35.8	40.4	38.1	39.7	39.5	38.7 ± 0.5
Coreset (k-center)	42.8	46.8	44.2	45.8	45.4	45.0 ± 0.8
EDGE	47.4	51.3	48.9	50.4	50.2	49.6 ± 0.5
Failure-only	42.2	46.4	43.5	45.2	44.9	44.4 ± 0.7
G-bandpass	44.9	48.8	46.3	47.7	47.5	47.0 ± 0.8
Random	39.5	43.7	41.2	42.7	42.4	41.9 ± 0.8
Recoverability	46.2	50.1	47.5	48.9	48.7	48.3 ± 0.5
Support-only	43.6	47.7	45.0	46.5	46.3	45.8 ± 0.4

For each PhysicalShift factor, we exclude that factor's records from scoring, querying, and correction training, and then evaluate on its disjoint resets. The named factor is an acquisition split label, not a feature supplied to the scorer. Nominal replay and the update budget remain fixed. The factor-specific training runs are first averaged within each experiment seed for the macro summary. Base is the unchanged base checkpoint evaluated on the same held-out resets, not an additional post-training run.

L QUERY-BUDGET CURVES

For adjacent mean-curve observations (B_0, s_0) and (B_1, s_1) bracketing a target $s^\star$, the illustrative crossing is

$$B^\star = B_0 + \frac{s^\star - s_0}{s_1 - s_0}(B_1 - B_0).$$

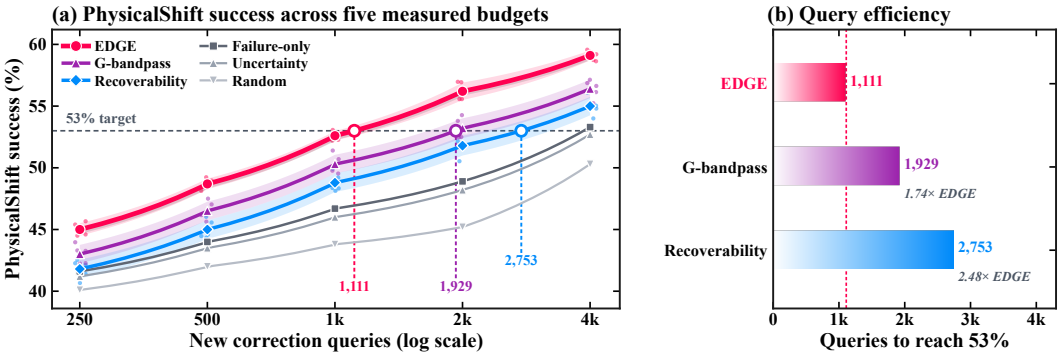

Figure 5: PhysicalShift budget sweep. (a) Means (markers) and sample deviations (bands) are reconstructed from the paired seed CSVs; small dots are seeds of the three strongest selectors. Lines are linear interpolation between measured budgets, so the marked 53% crossings are exactly the interpolated values given below. (b) Queries needed to reach the illustrative 53% target, by linear interpolation, and their ratio to EDGE. The 2,000-query records are shared with the main table.

Table 16: PhysicalShift success at five measured budgets. The 2,000-query rows are the main-experiment runs.

Queries	Failure-only	G-bandpass	Recoverability	EDGE
250	41.6 ± 0.6	43.0 ± 0.7	41.8 ± 0.8	45.0 ± 0.5
500	44.0 ± 0.7	46.5 ± 0.7	45.0 ± 0.8	48.7 ± 0.4
1000	46.7 ± 0.7	50.3 ± 0.8	48.8 ± 0.8	52.6 ± 0.4
2000	48.9 ± 0.9	53.2 ± 0.8	51.8 ± 0.8	56.2 ± 0.7
4000	53.3 ± 0.7	56.4 ± 0.7	55.0 ± 0.8	59.1 ± 0.4

The script does not extrapolate an unbracketed target. At $s^\star = 53\%$, the crossings are about 1,111, 1,929, and 2,753 queries for EDGE, *G*-bandpass, and Recoverability. The supplement also contains the Random and Uncertainty budget curves.

M Correction channels, compound shifts, and pools

M.1 Imperfect correction channels

Table 17: PhysicalShift correction-channel summaries. The deterministic reference row repeats the main seed records.

Correction channel	Failure-only	Support-only	EDGE	Recoverability
Main deterministic reference	48.9 ± 0.9	50.7 ± 0.8	56.2 ± 0.7	51.8 ± 0.8
5% action noise	47.7 ± 0.9	49.6 ± 0.9	54.8 ± 0.8	50.6 ± 0.8
10% action noise	46.0 ± 0.9	47.9 ± 0.9	52.8 ± 0.8	48.8 ± 0.9
1-step correction latency	45.6 ± 0.9	47.1 ± 0.9	51.9 ± 0.8	48.1 ± 0.9
half-horizon correction	46.4 ± 0.9	48.2 ± 0.9	52.7 ± 0.8	48.9 ± 0.9

The corrupted channels perturb the returned label through action noise, latency, or a shortened horizon, testing the robustness of a fixed acquisition rule to imperfect corrections.

M.2 Alternative planning-based oracle

A different correction source probes dependence on one controller, but a shared simulator, task predicates, and robot embodiment remain.

Table 18: Alternative motion-planning correction source. All selectors within this comparison use the same source.

Selector	PS success
Failure-only	46.2 ± 0.9
Support-only	48.0 ± 0.9
G-bandpass	50.4 ± 0.8
Recoverability	49.6 ± 0.8
EDGE	53.1 ± 0.7

M.3 Compound perturbations

Table 19: Separate compound-shift comparison; these conditions are not part of the main single-factor PhysicalShift score.

Compound shift	Selector	PS success
friction+delay	Base	31.4 ± 1.0
friction+delay	Failure-only	40.8 ± 1.0
friction+delay	G-bandpass	44.9 ± 0.9
friction+delay	EDGE	48.2 ± 0.8
mass+damping	Base	33.6 ± 0.9
mass+damping	Failure-only	42.4 ± 1.0
mass+damping	G-bandpass	46.7 ± 0.9
mass+damping	EDGE	50.1 ± 0.8

Compound conditions broaden the tested failure regimes but form a small, hand-picked set rather than systematic coverage of factor, severity, and task combinations.

M.4 Independent pool check

Table 20: Pool-specific outcome summaries. Each pool's standard deviation refers to its training-seed records.

Pool	Selector	PS success
Pool-A	Failure-only	48.9 ± 0.9
Pool-A	G-bandpass	53.2 ± 0.8
Pool-A	EDGE	56.2 ± 0.7
Pool-B	Failure-only	48.3 ± 1.0
Pool-B	G-bandpass	52.6 ± 0.9
Pool-B	EDGE	55.7 ± 0.8
Pool-C	Failure-only	49.4 ± 0.9
Pool-C	G-bandpass	53.8 ± 0.8
Pool-C	EDGE	56.8 ± 0.7

These records separate pool identity from within-pool training repetitions. Pool means should be compared at the pool level; treating every pool-by-training row as an independent acquisition pool would inflate the sample size.

N Compute accounting

The nominal models and scoring add $1.6 + 0.5 + 1.4 = 3.5$ GPU-hours, or 40.2% of the 8.7 GPU-hours of policy updating; when the nominal models are reused, scoring alone costs 1.4 GPU-hours (16.1%). Since G-bandpass also requires the nominal WAM ensemble, the marginal cost of EDGE over G-bandpass is the progress verifier and the candidate scoring, at most $0.5 + 1.4 = 1.9$ GPU-hours.

Table 21: Compute for one acquisition round. GPU-hours sum device use; elapsed times are rounded.

Stage	GPU-hours	Elapsed time on two A100-80GB devices
Nominal WAM ensemble	1.6	48 minutes
Nominal progress verifier	0.5	15 minutes
Deployment-pool scoring	1.4	42 minutes
Policy post-training, 5,000 steps	8.7	about 4.4 hours
Total per round	12.2	about 6.1 hours

O REPRODUCTION

The supplement contains seed-level outcome records, score-level acquisition replays, same-snapshot transport records with per-candidate rollback summaries, and per-reset binary downstream outcomes. Running `python -m pytest -q` from the supplement root executes 95 unit tests of the scoring, selection, and bandpass code, the analytic diagnostic, and the same-snapshot identities. `scripts/verify_artifact.py` and `scripts/verify_evidence.py` check integer counts, budget accounting, identifier separation, the selector replay, and the links between tables and records; further scripts regenerate the derived tables, the data figures, and the development-frozen bandpass validation (see the supplement `README`).

The released records reproduce every number in the reported tables and data figures. Raw camera streams, latent tensors, model checkpoints, and oracle action chunks are not included because of their size.

P SAME-SNAPSHOT PHYSICALSHIFT EVALUATION: PROTOCOL AND FACTOR RESULTS

Controlled comparison. Five paired seeds use all five PhysicalShift factors at moderate and severe levels, ten tasks, and forty logged snapshots per task–factor–severity cell, for 20,000 snapshots total. Each snapshot has one executed action and the same 15 alternative candidates under all three modes. The nominal WAM ensemble and progress verifier are frozen, and the confidence weight is fixed to one; the compared predictions are $\bar{F}(a)$, $\bar{F}(a) + e_t$, and $\bar{F}(a) + \rho(a, a_t)e_t$. Simulator rollback provides the continuation for every candidate.

For candidate k, the rollback reference difference is $q_k^* = V(z^*_{t+\Delta,k}) - v_t^+$, where v_t^+ is the verifier value of the observed continuation. The state reference teachability is

$$T^* = K^{-1} \sum_k \sigma((q_k^* - 0.05)/0.05),$$

exactly matching the operational form used by EDGE. The three predictors substitute their predicted candidate continuations into the same expression. Candidate latent MSE is measured directly against the rollback latent. State acquisition uses the shared G and $C = 1$, ranks GT, and purchases exactly 2,000 corrections. The reference ranking uses GT^*; overlap and regret are computed against that ranking, without looking at downstream policy success.

Table 22: PhysicalShift downstream success (%) when only residual transport changes, shown by evaluation factor and averaged over five paired seeds.

Factor	EDGE-N	EDGE-C	EDGE-K
Friction	52.0	55.4	56.5
Mass	52.0	55.4	56.4
Damping	52.0	55.1	56.3
Delay	51.8	55.6	55.4
Gain	52.3	55.8	56.4

The factor pattern matters more than an overall win. Kernel minus Constant is approximately $+1.1$, $+1.0$, $+1.2$, -0.2, and $+0.6$ pp for friction, mass, damping, delay, and gain, respectively. Delay is the expected failure case for zeroth-order local transport because a discrete control-step shift can change which actuation reaches the environment. Gain is intermediate: local anchoring helps, but a single residual cannot fully repair the changed action-response Jacobian.

Released records. `physicalshift_transport_candidates.csv` contains the common snapshot/candidate keys, rollback progress, per-mode candidate differences, and latent MSE. `physicalshift_transport_snapshots.csv` contains the exact soft-T values and acquisition scores. Selection files provide the paired top-2k metrics; downstream seed/factor tables are exact aggregates of `physicalshift_transport_downstream_reset_outcomes.csv`. `verify_same_snapshot.py` enforces the executed-action identity and recomputes q^*, soft-T, exact ranking metrics, pairing, budgets, binary success counts, and factor-wise contrasts from those records. These scalar records reconstruct every number in Tables 4 and 22.

Q ANALYTIC RESIDUAL-TRANSPORT CONTROLS

Purpose. This analytic control complements the PhysicalShift rollback experiment of Appendix P. Under known transition functions it separates the observed offset from its distance-dependent decay, exposing positive and negative cases for the locality assumption. The protocol, generator, primitive arrays, per-snapshot metrics, seed aggregates, paired contrasts, and execution ledger are in `diagnostics/transport/`.

Known transition model. Each of 20 seeds creates $N = 256$ states in $\mathbb{R}^8$ and $K = 16$ actions in $\mathbb{R}^4$. The executed action is $a_0 = 0$; each other action has independent uniform radius in $[0.15, 2]$ and a normalized Gaussian direction. Nominal next-state prediction is $z + Wa$, with independent

$z_j \sim \mathcal{N}(0, 0.5^2)$ and $W_{ij} \sim \mathcal{N}(0, 0.45^2)$. State-specific $b_{0j} \sim \mathcal{N}(0, 0.55^2)$ defines an offset. The ground-truth next state is $z + Wa + b(a)$, where

$$b(a) \in \left\{ b_0,\ e^{-\|a\|^2/(2\ell^2)} b_0\ (\ell = 0.5, 1, 2),\ b_0 + Ja,\ 0 \right\}, \qquad J_{ij} \sim \mathcal{N}(0, 0.2^2).$$

The observation at a_0 adds noise $s\epsilon$, with $\epsilon \sim \mathcal{N}(0, I)$ and $s \in \{0, 0.05, 0.2\}$. The same primitive states, actions, noise vectors, and nominal predictions are used for all methods and discrepancy conditions within a seed. The Nominal, Constant, and Kernel predictors add 0, e, and $\exp(-\|a\|^2/2)e$ to the nominal next state, respectively. The kernel length is always 1 and $C = 1$ for every method.

Progress and metrics. A fixed normalized random w, with $\|w\| = \sqrt{d}$, defines two bounded heads:

$$V_{\text{lin}}(u) = \sigma(w^\top u/\sqrt{d}), \qquad V_{\text{rad}}(u) = \exp[-\|u - w\|^2/(2d)].$$

We report latent mean-squared error and progress absolute error on the 15 nonexecuted candidates. Candidate regret is $\max_k V(z_k^*) - V(z_{\hat{k}}^*)$, with $\hat{k} = \arg\max_k V(\hat{z}_k)$ over all 16 candidates, including the executed one; ties use candidate order. Top-1 accuracy is also released. Metrics are first averaged inside each seed. Differences use paired seeds; the CSV contains unadjusted 95% t intervals for every method contrast.

Table 23: Analytic test at observation-noise standard deviation 0.05 with the nonlinear radial progress head. MSE is mean per latent coordinate; regret is multiplied by 100 and shown as mean±sample SD across 20 seeds. Smaller is better. N/C/K denote nominal, constant-residual, and kernel-residual prediction. Results at all noise levels are in the supplement.

Discrepancy	MSE N	MSE C	MSE K	Regret N	Regret C	Regret K
Constant offset	0.3052	0.0025	0.0820	1.71 ± 0.29	0.01 ± 0.01	1.09 ± 0.14
Local, $\ell = 0.5$	0.0491	0.1992	0.0346	1.20 ± 0.70	2.11 ± 0.47	0.53 ± 0.14
Local, $\ell = 1$ (matched)	0.1208	0.0836	0.0010	1.61 ± 0.60	1.26 ± 0.27	0.01 ± 0.00
Local, $\ell = 2$	0.2216	0.0143	0.0325	1.60 ± 0.36	0.28 ± 0.07	0.41 ± 0.07
Affine discrepancy	0.3625	0.0601	0.1395	3.44 ± 0.40	1.75 ± 0.37	2.95 ± 0.50
Zero bias	0.0000	0.0025	0.0010	0.00 ± 0.00	0.02 ± 0.01	0.02 ± 0.01

Interpretation. Constant transport wins for a truly constant discrepancy. The narrow and matched local regimes favor kernel transport over constant transport, whereas the broader and affine regimes need not. The matched Gaussian deliberately equals the assumed kernel and is only a positive control. The zero-bias regime tests whether transporting pure observation noise is harmful. These cases illustrate conditional behavior.

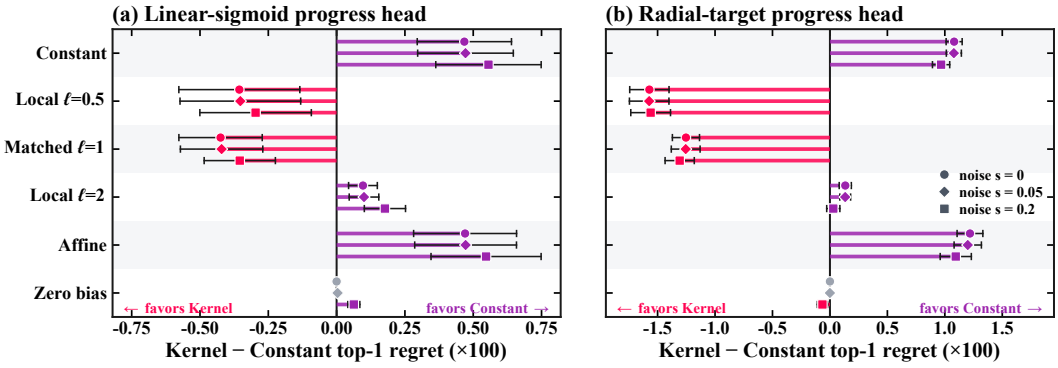

Figure 6: Kernel-minus-constant top-1 regret, multiplied by 100, for every discrepancy regime at all three observation-noise levels (marker shape). Below zero favors kernel transport; above zero favors constant transport. Whiskers are paired, unadjusted 95% t intervals across 20 seeds. The two heads use the same transitions.

Why two progress heads? For a sigmoid of a linear projection, a shared constant latent shift does not change candidate order; nominal and constant transport therefore have identical ranking in that

head even when their prediction errors differ. Because of this invariance, we add a nonlinear radial head and report both heads for all seeds and conditions. The invariance also shows that lower latent MSE need not imply better corrective-action ranking.

Table 24: The same diagnostic with the linear-sigmoid progress head. Nominal and constant regret coincide because of the ordinal invariance explained above. Column definitions match Table 23.

Discrepancy	MSE N	MSE C	MSE K	Regret N	Regret C	Regret K
Constant offset	0.3052	0.0025	0.0820	0.00 ± 0.00	0.00 ± 0.00	0.47 ± 0.37
Local, $\ell = 0.5$	0.0491	0.1992	0.0346	0.63 ± 0.63	0.63 ± 0.63	0.27 ± 0.20
Local, $\ell = 1$ (matched)	0.1208	0.0836	0.0010	0.42 ± 0.33	0.42 ± 0.33	0.00 ± 0.01
Local, $\ell = 2$	0.2216	0.0143	0.0325	0.08 ± 0.05	0.08 ± 0.05	0.18 ± 0.15
Affine discrepancy	0.3625	0.0601	0.1395	1.81 ± 0.93	1.81 ± 0.93	2.28 ± 1.25
Zero bias	0.0000	0.0025	0.0010	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00

Reproduction. Running `run_transport_diagnostic.py` generates the inputs and all 2,160 seed–head–noise–regime–method summary rows. The `--verify` mode regenerates and compares every input array, per-snapshot metric array, aggregate, and protocol binding. There are 92,160 state–regime–noise combinations over 5,120 generated base states, with repeated measures across heads and methods.

R DEVELOPMENT-FROZEN SIMPLE BASELINE

We tune the center of $s_c(G) = -|G - c|$ only on development tasks, using the same correction budget, exclusion window, and policy-update schedule. The grid is $\{0.05, 0.10, \ldots, 0.95\} \cup \{0.68\}$. The chosen center is $c = 0.75$; it is frozen before the ten paired validation seeds are evaluated. Development and validation task identifiers are disjoint, and the implementation rejects missing cells, duplicate runs, noninteger success counts, unequal evaluation budgets, and split overlap.

The complete 20-center development grid is released as binary outcomes and integer aggregates. `freeze_bandpass.py` recomputes the development winner and checks it against the hash-stamped freeze record of the development files. On the ten validation seeds, EDGE averages 56.16% PhysicalShift success and the frozen $c = 0.75$ bandpass averages 53.14%, a mean paired margin of approximately $3.0\,\mathrm{pp}$. The release includes per-reset binary outcomes for both methods at five correction budgets (250, 500, 1,000, 2,000, 4,000); the 2,000-query cell exactly reconstructs `bandpass_validation_seed_results.csv`.