WHEN POLICY REPRESENTATIONS FAIL: DIAGNOSIS AND REPAIR THROUGH BEHAVIORAL EVIDENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Compact policy representations adapt to downstream tasks by optimizing low-dimensional coordinates rather than the full policy parameter vector, reducing adaptation overhead. However, these representations are typically kept fixed after training. When adaptation to a new task fails, expanding the representation is a natural response. We show that such failure is only a symptom, not evidence of insufficient capacity: similar performance degradation can result from limited target-relevant information, inadequate coordinate search, poor organization of behavior within the existing representation, or genuinely missing adaptation directions. We prove that final performance alone is insufficient to uniquely identify the source of failure or determine how the representation should be modified. Because these latent factors are not directly observable, we introduce a behavioral witness: an independently verified successful target adaptation that provides both behavioral and directional evidence. Building on this reference, we develop a witness-guided hierarchical diagnosis and repair framework. Under fixed budgets, the framework tests pre-specified interventions from weaker to stronger and selects the weakest verified repair that satisfies both target-performance and behavior-preservation requirements, expanding the representation only when fixed-capacity alternatives remain insufficient. We evaluate the framework in continuous control and vision-language-action policies. The results show that behavioral evidence supports different levels of sufficient repair, turning default expansion into an evidence-guided, on-demand process that avoids unnecessary representation growth. At the repair stage, the resulting approach also substantially reduces trainable-parameter and peak-memory overhead relative to full fine-tuning.

1 INTRODUCTION

Compact policy representations let new tasks adapt through low-dimensional coordinates rather than full policy parameters. FTB (From Parameters to Behaviors) learns a latent policy manifold organized by behavioral similarity (Tenedini et al., 2026). PG-ELLA and LPG-FTW update shared policy bases as tasks arrive (Ammar et al., 2014; Mendez et al., 2020), while Progress & Compress consolidates new knowledge into a shared model (Schwarz et al., 2018). These methods address representation construction and reuse. We ask an earlier question: **when adaptation within a compact representation fails, which change is sufficient to recover performance?**

Expansion is a natural response, as in progressive and dynamically expandable continual learning (Rusu et al., 2016; Yoon et al., 2018). But low return is a symptom, not evidence of missing capacity. Target-relevant information may be absent, coordinate search may miss an available policy, or the existing map may organize behavior poorly. A task may instead require genuinely new directions. These possibilities call for different interventions: stronger search, information repair, same-capacity remapping, or expansion. Final performance alone cannot distinguish them.

Consider two adaptations with equally poor target return. In one, an effective policy is already reachable, but optimization stops at an ineffective coordinate. In the other, the current map cannot reach the desired behavior. Expanding both representations may improve their observed return, yet the first case could have been resolved without changing the map. Even in the second case, a different map with the same capacity may suffice. The practical decision is therefore how much of the representation needs to change, rather than whether an added direction can improve performance.

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.

Our analysis separates representation loss from finite-budget search error. A fixed-capacity bound further depends on source–target discrepancy, representation fit, and best target approximation. Different combinations produce the same gap. These population-level quantities are not directly observable, motivating intervention tests rather than a unique causal diagnosis from return.

We use a **behavioral witness**: an independently verified adaptation with positive target improvement. Its actions describe effective behavior, and its parameter change supplies a task-relevant direction. Repairs are tested against these references under matched budgets; failure to fit this witness does not prove the task impossible within the representation.

The candidate and the repaired representation answer different questions. A successful candidate demonstrates that a more flexible adaptation can improve the target. It can be used directly when its storage and adaptation requirements are acceptable. Here, it also serves as evidence for a restricted repair: can its useful behavior be accessed by changing coordinates, absorbed into existing parameters, or recovered through an added direction? The repaired policy is trained and evaluated under its own constraints. Candidate performance supplies a reference, rather than being counted as the repair's outcome.

Our **witness-guided diagnosis and repair framework** tests predeclared interventions from weaker to stronger: coordinate search, behavior-informed repair, same-capacity mapping and their combinations, then witness-aligned expansion. It selects the weakest verified repair meeting target-performance and behavior-preservation requirements. This identifies sufficiency within the tested family and budget, not a unique cause or universally necessary repair order.

The hierarchy concerns the commitment made by an intervention. Search keeps the map fixed; information and mapping repairs retain the declared coordinate and decoder capacity; expansion adds a persistent adaptation direction. Joint tests allow limitations to coexist instead of forcing every case into a single category. Testing fixed-capacity alternatives first makes growth conditional on their observed insufficiency. The original representation is retained whenever a weaker tested change already meets the target and archive requirements.

Controlled constructions test the diagnostic distinctions. Natural continuous-control tasks exhibit single-level and joint sufficient repairs, while vision-language-action adaptation tests witness directionality in a different representation. Matched controls separate task-relevant evidence from additional optimization and arbitrary capacity. Shared-candidate continuation also reduces repair-stage trainable parameters and peak memory relative to full fine-tuning; candidate generation remains a separate cost.

2 Related Work

Compact and lifelong policy representations.
Compact policy representations exploit low-dimensional structure to support downstream adaptation. FTB learns a behaviorally structured latent policy manifold, while PG-ELLA and LPG-FTW maintain shared latent bases or factored policy representations across tasks (Ammar et al., 2014; Mendez et al., 2020; Tenedini et al., 2026). These methods focus on learning and maintaining shared representations. Our setting begins once such a representation has already been learned and adaptation within it fails.

Capacity expansion and representation reuse.
Continual-learning methods may accommodate new tasks by adding task-specific capacity, as in progressive or dynamically expandable networks (Rusu et al., 2016; Yoon et al., 2018), or by reusing and consolidating existing knowledge. Progress & Compress distills newly acquired knowledge back into a fixed-capacity model, while AdapterFusion and LoRAHub compose previously learned adapters or LoRA modules (Schwarz et al., 2018; Pfeiffer et al., 2021; Huang et al., 2024). These alternatives motivate our focus on testing whether a nonexpanding repair is already sufficient before persistent capacity is added.

Low-rank adaptation and task directions.
O-LoRA and InfLoRA use constrained low-rank subspaces for continual adaptation, while Task Arithmetic and TIES-Merging study task-specific parameter directions and their composition (Wang et al., 2023; Liang & Li, 2024; Ilharco et al., 2023; Yadav et al., 2023). Sequential LoRA has also

been shown to be a strong VLA continual-adaptation baseline (Hu et al., 2026). In our framework, witness-derived task directions serve as diagnostic evidence within a repair hierarchy: fixed-capacity alternatives are evaluated first, and task-aligned expansion is introduced only when they remain insufficient.

3 Hierarchical Diagnosis and Repair

3.1 Representation Failure Decomposition

Let $J_\tau(\theta)$ be the return of policy π_θ on task τ. A compact map $G_\phi : \mathcal{Z} \to \Theta$ restricts adaptation to coordinates z. The unconstrained and chart-reachable optima are

$$J_\tau^\star = \sup_{\theta\in\Theta} J_\tau(\theta), \qquad V_\tau(\phi) = \sup_{z\in\mathcal{Z}} J_\tau(G_\phi(z)).$$

Following the distinction between approximation and optimization error (Bottou & Bousquet, 2008), a fixed-budget solution $\hat{z}$ has gap

$$J_\tau^\star - J_\tau(G_\phi(\hat{z})) = \underbrace{J_\tau^\star - V_\tau(\phi)}_{L_\tau(\phi)} + \underbrace{V_\tau(\phi) - J_\tau(G_\phi(\hat{z}))}_{O_\tau(\phi,\hat{z})}. \tag{1}$$

L_τ measures unreachable performance; O_τ measures search error within the chart. Search error alone does not require representation change.

Following domain-adaptation bounds (Ben-David et al., 2010), fix a class Φ_c with specified latent dimension, decoder, permitted updates and archive constraints. For its source-trained map $\hat{\phi}$, let $R_{\mathrm{src}}(\phi) = \mathbb{E}_{\tau'\sim P_{\mathrm{src}}}[L_{\tau'}(\phi)]$ and define

$$C_\tau = \sup_{\phi\in\Phi_c} |L_\tau(\phi) - R_{\mathrm{src}}(\phi)|, \qquad M_{\mathrm{src}} = R_{\mathrm{src}}(\hat{\phi}) - \inf_{\phi\in\Phi_c} R_{\mathrm{src}}(\phi), \qquad K_\tau = \inf_{\phi\in\Phi_c} L_\tau(\phi).$$

C_τ measures source–target discrepancy, M_{src} source mapping regret, and K_τ the best target approximation error in Φ_c. Then

$$L_\tau(\hat{\phi}) \le K_\tau + M_{\mathrm{src}} + 2C_\tau. \tag{2}$$

These potentially coexisting limitations motivate Search, Information, Mapping and Capacity tests; return alone cannot distinguish them. Appendix A derives the bound and its relation to prior theory.

Expansion can reduce finite-budget search error even when the best reachable return is unchanged. An observed expansion gain therefore does not isolate approximation loss. Conversely, a failed fixed-capacity optimizer provides a budget-specific outcome, not the class-wide infimum defining K_τ. We use the decomposition to identify which comparisons are needed: keep the map fixed to test search, retain capacity while varying evidence or permitted map updates, and test added directions against matched controls.

3.2 Behavioral Witness as Diagnostic Evidence

Since class-wide optima are unavailable, we test repairs against a successful adaptation. A flexible operator produces $\theta_c = F_\tau(\theta_0)$ from parent $\theta_0 = G_{\hat{\phi}}(\hat{z})$. In MuJoCo, certification as θ_w requires a frozen screen and independent paired improvement:

$$\widehat{\Delta}_c = \widehat{J}_\tau(\theta_c; \mathcal{D}_{\mathrm{eval}}) - \widehat{J}_\tau(\theta_0; \mathcal{D}_{\mathrm{eval}}) > 0. \tag{3}$$

Eight candidates receive 512 interactions each. Screening uses eight paired 256-step episodes, requiring positive mean gain and at least five positive pairs; the selected candidate receives 64 independent paired episodes, with nonpositive gain triggering abstention. Training, screening and evaluation samples are disjoint; parent/candidate evaluation seeds match. Tracking accepts $12/12$ witnesses, and unseen-target screening accepts $11/12$ with one abstention. This is a finite-sample gate, not a per-case confidence certificate. Behavior $b(\theta_w)$ supports fitting; displacement $\theta_w - \theta_0$ supports direction tests. Fit, inversion and audit states are disjoint (Appendices B, H).

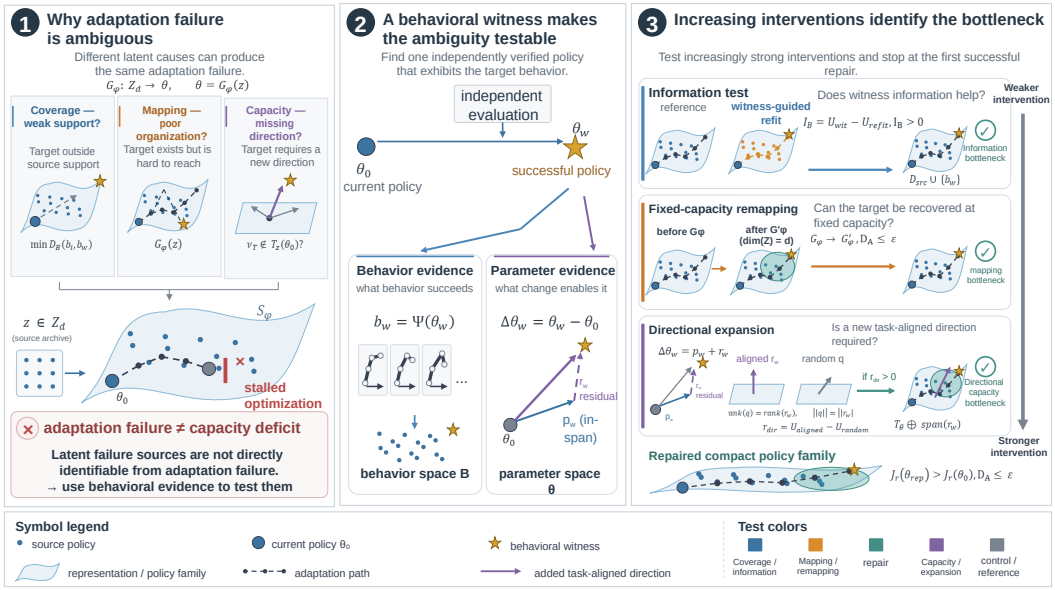

Figure 1: **Witness-guided diagnosis and repair.** A verified adaptation supplies behavioral and parameter evidence for progressively stronger interventions. The sketches represent hypotheses. Natural MuJoCo uses witness-conditioned fitting and a candidate-minus-base secant; VLA uses chart-projected residuals and tests directionality without a per-target witness gate.

3.3 Hierarchical Interventional Diagnosis

Interventions start separately from the same parent. We cross archive-only evidence D_0 versus witness-augmented evidence D_1 with encoded coordinates A_0 versus behavior inversion A_1, matching sample budget, latent dimension and patch structure. Inversion uses a 33×33 grid and eight 128-step gradient starts, optimizing only the code, not the decoder relative to A_0 at the same D_i:

$$U_{ij} = U_\tau(A_j(D_i)), \qquad i, j \in \{0, 1\}. \tag{4}$$

Fitting and inversion may interact. We test both fit-then-invert and invert-then-fit paths (Appendix C.3); inversion itself is Search, not decoder Mapping.

The tested intervention sets are nested:

$$\mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \mathcal{F}_2 \subseteq \mathcal{F}_3, \tag{5}$$

$\mathcal{F}_0$ contains stronger searches; $\mathcal{F}_1$ adds witness-conditioned fixed-capacity fitting; $\mathcal{F}_2$ adds declared joint/reverse paths; $\mathcal{F}_3$ adds aligned expansion. These finite families differ between tracking and later path experiments and do not exhaust each model class. With shared archive constraints and target requirement u_{req}, select

$$k^\star = \min\left\{k : \exists \pi \in \mathcal{F}_k,\ J_\tau(\pi) \geq u_{\text{req}},\ D_{\text{archive}}(\pi) \leq \epsilon_A\right\}. \tag{6}$$

The selected $k^\star$ identifies tested sufficiency, not a unique cause. Appendix C gives the factorial effects and finite-budget rule.

Each test is defined by what changes. Search optimizes coordinates while holding decoder tensors fixed. Information repair changes the action targets used to fit a preallocated patch. The separate decoder Mapping test changes whether that patch remains trainable during downstream PPO. Thus fixed capacity means unchanged allocated structure, not unchanged weights. The Information–Mapping factorial evaluates their separate and joint effects, while fitting–inversion path tests evaluate the order of different operations. These distinctions prevent a successful latent inversion from being labeled decoder repair.

3.4 Diagnosis through Targeted Repair

Fixed-capacity repair fits witness trajectories with archive replay. Archive-only refitting matches architecture, initialization and update budget, isolating witness information without adding latent coordinates.

After the strongest *tested* fixed-capacity intervention fails, let $\bar{G}$ be its map, $\theta_b = \bar{G}(z_b)$ its target policy, and $d_w = \theta_w - \theta_b$. The idealized chart-external component is

$$r_\perp = \left(I - P_{\mathrm{col}(J_{\bar{G}}(z_b))}\right) d_w. \tag{7}$$

A nonzero $r_\perp$ may reflect curvature, parameter symmetries or behaviorally irrelevant changes. The strict post-mapping FTB test uses the *unprojected* secant from trained U_{11}; later confirmation uses U_{01}. VLA truncates a source-projected residual, which need not remain orthogonal after truncation, and matches random controls in layer-wise rank and spectrum. Repaired policies are evaluated separately from the witness (Appendices D, H, I).

3.5 Instantiations

MuJoCo uses a nonlinear FTB decoder and PPO candidates (Todorov et al., 2012; Schulman et al., 2017). SmolVLA uses source-task LoRA adapters and a target LoRA displacement (Shukor et al., 2025; Hu et al., 2022), testing directionality without the full hierarchy or per-target witness gate. Appendices H–J specify both implementations.

4 Experiments

MuJoCo tests repair selection and preservation; SmolVLA/LIBERO tests directionality in another representation (Liu et al., 2023). Following FTB and Progress & Compress, we evaluate return and behavioral preservation (Tenedini et al., 2026; Schwarz et al., 2018). Matched refitting and random directions isolate information and alignment; factorial interventions separate single and joint repairs (NIST/SEMATECH, n.d.). Exact results and configurations appear in Appendix K.

4.1 Controlled Identifiability of Representation Failures

Linear examples first distinguish coordinate Search from Mapping. We then plant Coverage, Mapping and Capacity limitations with matched initial error and archived behavior. The tests recover these planted distinctions in rotated constructions and trained FTB charts across HalfCheetah, Walker2d and Hopper (Table 1).

Skipping Information or Mapping sends repairable cases to Capacity prematurely; retaining both checks avoids escalation in either inspection order (Figure 2). Replay compares offline decisions, not executed expansions or rollout success (Table 14).

Planted constructions make the intended limitation known, allowing selection to be scored against a controlled reference. Natural tasks instead ask which tested intervention meets the acceptance criteria without access to a causal label. We therefore use controlled tests to check the diagnostic distinctions and natural evaluations to check repair sufficiency. Keeping these roles separate avoids interpreting the planted classification result as natural-task diagnosis accuracy.

4.2 Mechanism Validation of Behavioral and Directional Evidence

Witness repair and archive-only refitting share the patch, archive, optimizer and budgets. A second comparison matches added dimension and direction norm while changing alignment. Both favor task-relevant evidence on the frozen MuJoCo panel (Table 2); VLA additionally matches layer-wise rank and spectrum.

In the complete prospective subset, witness repair lowers cost relative to refitting, and aligned expansion improves on random directions (Figure 3; Table 3). Cost is negative return per horizon step. The panels compare separate interventions, not training stages.

4.3 Hierarchical Diagnosis on Natural MuJoCo Tasks

Targeted HalfCheetah rewards were chosen after development inspection and frozen before candidate outcomes. All candidates improve independent return; passing requires the predeclared gain-recovery and fixed-code archive criteria (Table 16).

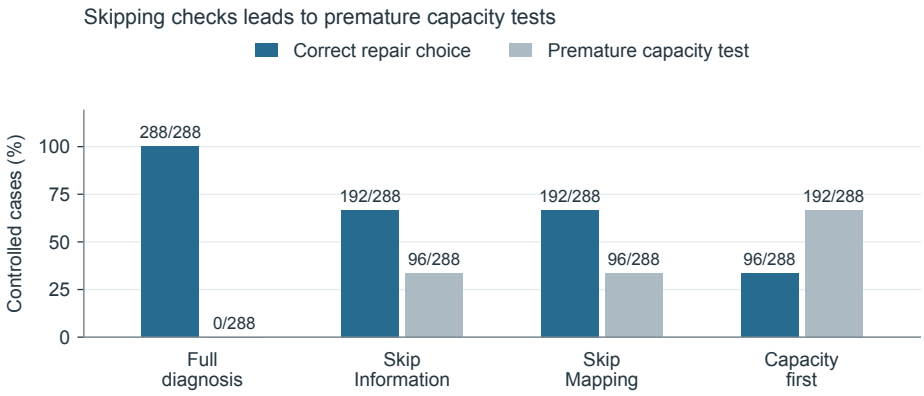

Figure 2: **Keeping both checks prevents premature escalation.** Blue: correct choices; gray: premature Capacity tests on the same controlled panel. These are offline decisions, not rollout successes. Counts and configurations appear in Table 14.

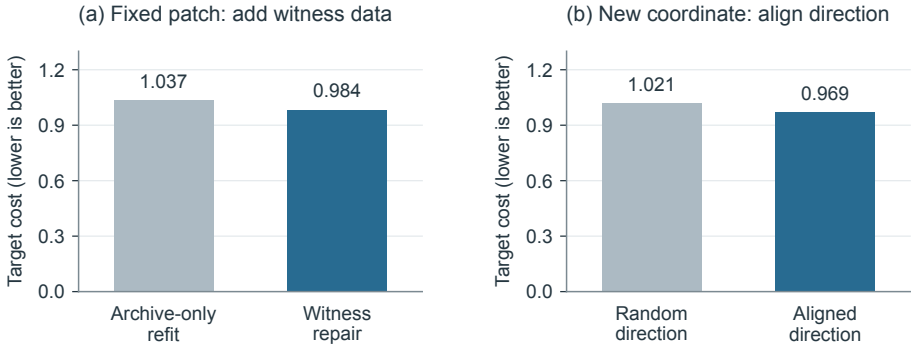

Figure 3: **Behavioral information and alignment reduce target cost.** Means over the complete prospective HalfCheetah panel under matched budgets. (a) Shared patch and archive; (b) equal dimension and norm. Lower is better. Complete results appear in Tables 2 and 3.

Cases pass through single fixed-capacity paths, combined fitting and inversion, or aligned expansion. Because path memberships overlap, the cohort summary separates raw counts from first sufficient selection (Figure 9). These joint paths involve code inversion, not decoder updates.

Figure 4 distinguishes Search, Information, actual decoder-only repair, Capacity and joint repair. It compares independent arms, not stages. Appendix K.4 gives selection rules, measurements and complete cohort counts.

The case comparisons show why the intervention identity matters. Stronger Search succeeds without moving the decoder, whereas the Information example needs witness-conditioned fitting despite unsuccessful search and inversion. In the decoder example, both decoder-only and joint repair pass, making the decoder update a sufficient tested choice. The joint-only example has a different pattern: neither single intervention reaches the criterion. Its evidence is the full factorial contrast, rather than improvement of the joint arm over no repair alone. Because these examples come from distinct cohorts, their bar heights are read within each panel, not as a common learning trajectory.

Retaining inversion and joint checks avoids unnecessary Capacity tests; exchanging the Information and inversion checks leaves replay selections unchanged. Actual repair order matters on a separate panel: invert-then-fit succeeds where fit-then-invert fails under fresh evaluation. Another panel confirms information repair without new coordinates (Tables 13 and 15).

Original tracking records equal pass rates for selection and always-expand, with fewer expansions but more interactions. Its logged archive reference is incomplete (Table 20). A later control verifies Search with the untouched decoder (Figure 10, Table 17).

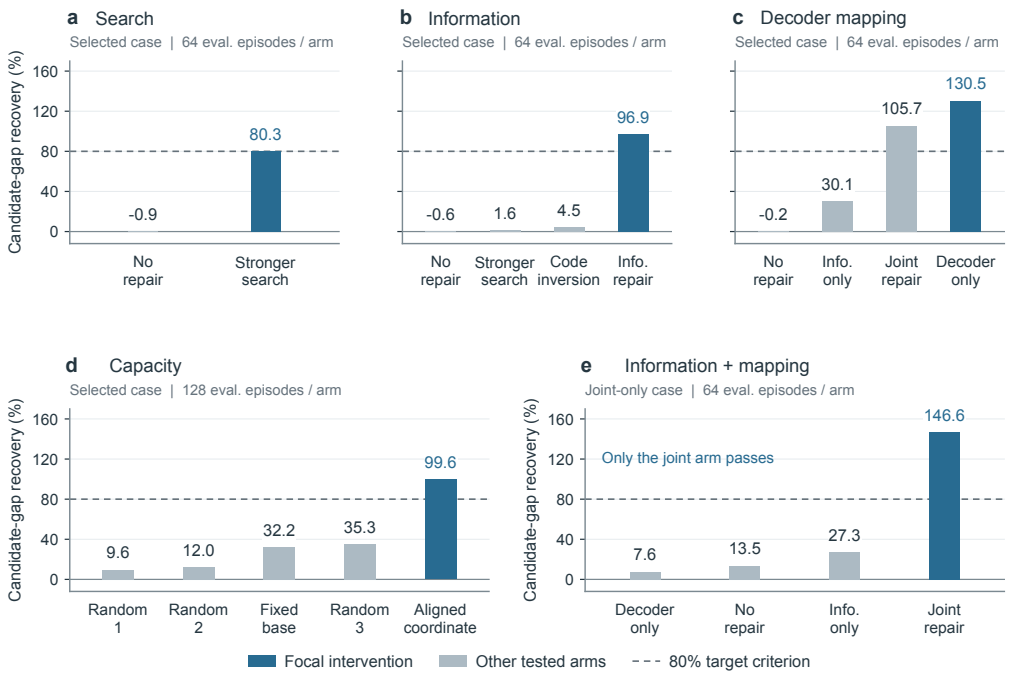

Figure 4: **Single and joint repairs in natural examples.** (a,b) Search and Information; (c) decoder updates; (d) aligned versus random directions; (e) joint-only repair. Bars recover candidate–parent gain; dashed line: target requirement. All arms meet the archive criterion; blue marks the focal arm. Examples use different cohorts and protocols, sorted by score, not stage (Appendix K.4).

4.4 New-Task Confirmation of Capacity and Joint Repair

Capacity and true decoder repair share a development-disjoint reward panel, frozen before candidate outcomes. Capacity screens starting checkpoints for archive preservation; qualified and unqualified cases remain in Appendix K.3.

With old code and decoder frozen, we add a scalar coordinate along either the candidate-minus-base direction or an equal-norm random direction. Every coefficient starts at $\alpha = 0$ and receives identical candidate-action write-in and PPO budgets. Aligned expansion passes all archive-qualified cases, while random controls seldom pass (Figure 5a; Table 4).

Resetting the trained coefficient recovers audited old-code behavior, not old-task rollout retention (Table 5). The zero start precedes behavioral write-in, rather than removing supervision.

Starting coefficients at zero gives the trained direction arms the same initial policy; matched random controls test directionality during write-in and adaptation. This unconditional mechanism test includes cases already resolved by fixed-capacity repair, so it does not establish expansion necessity or evaluate nested selection. Resetting the coordinate tests recovery of archived actions against the original archive, including any drift already present in the starting checkpoint.

We cross archive-only or witness-conditioned initialization with frozen or trainable decoder patches, without adding coordinates. A confirmation case passes only under joint repair (Figure 5b). Tables 6 and 7 give the factorial and measurements.

Information and joint repair resolve equal numbers overall, so the joint-only case establishes existence rather than prevalence or average benefit. Gain recovery can exceed the candidate reference; it is not success probability.

4.5 Cross-Representation Validation in SmolVLA

In SmolVLA, source LoRA adapters define the chart and a target LoRA supplies the candidate. We truncate the candidate-minus-projection residual and compare it with controls matched in layer-

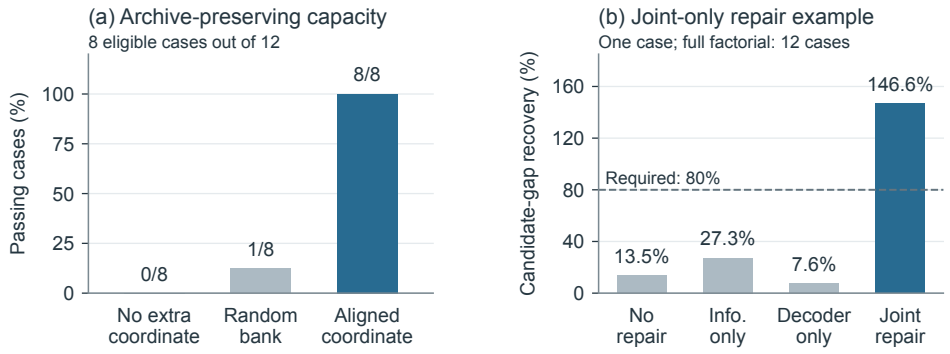

Figure 5: **Capacity and joint confirmation.** (a) Archive-qualified cases under matched aligned/random budgets; the base reference receives no additional training. Old behavior is audited at coefficient zero. (b) Only joint repair meets the target criterion in the displayed case. Complete counts and audits appear in Appendix K.3.

wise rank and spectrum but randomized in orientation. This tests directional information, not strict orthogonality of the truncated update.

On frozen Random-20, Aligned retains much of Direct's pooled success count, while matched random residuals fail throughout the panel. The retained-count ratio is not episode-wise success retention (Table 23). Figure 6 also tests residual rank on a separate panel. Actual simulator sequences are shown separately in Appendix I.2.

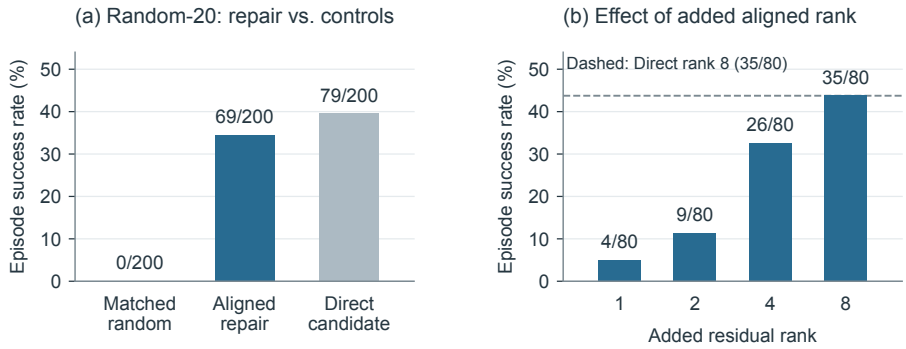

Figure 6: **Aligned repair in simulated manipulation.** (a) Frozen Random-20 success counts; Direct is the candidate reference. (b) Residual-rank sweep on a separate target panel; dashed line: Direct. Both plots report complete panels, with exact results in Tables 23 and 27. Simulator examples are displayed separately in Figure 8.

4.6 Robustness and Practical Performance

Aligned residuals also succeed on the held-out four-suite panel and rebuilt source-qualified chart, while matched random remains at zero (Tables 24 and 26). The untruncated four-suite construction reconstructs Direct algebraically; its rollout discrepancies remain unexplained and are not evidence of superiority over Direct (Appendix I.1). Figure 6 gives the residual-rank sweep.

Coefficient-learning controls show that direction alone does not recover learning under ordinary PPO. Development tuning does not establish an aligned advantage, whereas matched candidate-action write-in before PPO separates aligned from random directions. Behavior fitting is therefore needed under these budgets (Table 18).



Figure 7: **A recorded Aligned rollout in LIBERO.** Initial, middle and final frames of the first logged successful episode for the first Random-20 task: placing ketchup in a basket. This selected example illustrates behavior; aggregate results use the full panel. Unaltered frames, selection details and a second task appear in Appendix I.2.

In shared-candidate continuation, Aligned exceeds Full FT and Direct LoRA in pooled success while reducing trainable parameters, peak memory and continuation-plus-construction time (Tables 31 and 32). These comparisons cover continuation from a common candidate, not end-to-end adaptation from the pretrained model.

Fixed-storage continual reuse remains unreliable: final success is low and the AUC advantage reverses across task orders. A task-routed adapter library performs better, but its storage grows with the task count. Table 29 reports complete retention and storage results.

These comparisons separate adaptation overhead from representation storage and task reuse. Freezing an existing chart reduces optimizer-visible weights and their memory cost, but does not remove the stored chart or witness acquisition. Likewise, a successful local repair need not yield a single compact adapter that retains a sequence of tasks. The continuation and continual-learning protocols measure these different properties separately, so local repair gains are not interpreted as evidence of reliable fixed-storage consolidation.

5 Limitations

Natural-task experiments establish sufficient repairs at all four tested levels and a joint-only confirmation, but do not estimate their prevalence or agreement across successful witnesses. The reward panels cover HalfCheetah, with the targeted panel chosen after development inspection. Branches use point estimates without per-case confidence guarantees. Capacity recovery is audited with the new coefficient reset; old-task rollout retention remains untested. Witness generation and diagnosis incur costs. SmolVLA tests directionality rather than the full hierarchy; truncation leakage and full-residual rollout equivalence remain unaudited at checkpoint level. Fixed-storage continual reuse remains unreliable.

6 Conclusion

Behavioral witnesses make compact-policy repair a testable choice rather than a default expansion. Natural MuJoCo experiments exhibit sufficient Search, Information, decoder Mapping, and Capacity interventions. Development-disjoint confirmation shows aligned expansion passing all eight archive-qualified cases with recoverable old-code behavior, and a mixed case resolved only by joint information and decoder repair. SmolVLA supports candidate-aligned directions over matched random capacity; shared-candidate continuation also reduces trainable-parameter and peak-memory overhead. Together, these results support selecting the weakest verified sufficient repair within the declared family, testing fixed-capacity alternatives before adding task-relevant directions.

AI USE STATEMENT

Generative AI assisted research discussion, record organization, code, drafting, and editing (Kassis et al., 2026). The authors remain responsible for checking code, results, citations, and claims against archived artifacts.

ETHICS STATEMENT

Evaluations use simulated control and existing robot demonstrations, with no new human-subject data or autonomous deployment procedure for physical robots.

REPRODUCIBILITY STATEMENT

Appendix K records protocols, complete outcomes, and costs. The package includes LaTeX, figures, plotting inputs, and rollout sources. Training code, checkpoints, and data processing are separate; the manuscript package alone does not reproduce training.

REFERENCES

Haitham Bou Ammar, Eric Eaton, Paul Ruvolo, and Matthew E. Taylor. Online multi-task learning for policy gradient methods. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 1206–1214. PMLR, 2014. URL https://proceedings.mlr.press/v32/ammar14.html.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1–2): 151–175, 2010. doi: 10.1007/s10994-009-5152-4. URL https://doi.org/10.1007/s10994-009-5152-4.

Léon Bottou and Olivier Bousquet. The tradeoffs of large scale learning. In J. C. Platt, D. Koller, Y. Singer, and S. Roweis (eds.), *Advances in Neural Information Processing Systems 20 (NIPS 2007)*, pp. 161–168. NIPS Foundation, 2008. URL https://leon.bottou.org/papers/bottou-bousquet-2008.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Jiaheng Hu, Jay Shim, Chen Tang, Yoonchang Sung, Bo Liu, Peter Stone, and Roberto Martín-Martín. Simple recipe works: Vision-language-action models are natural continual learners with reinforcement learning. In *Reinforcement Learning Conference*, 2026. URL https://arxiv.org/abs/2603.11653.

Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. LoRAHub: Efficient cross-task generalization via dynamic LoRA composition. In *Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=TrloAXEJ2B.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Timothy Kassis, Vinayak Agarwal, Yuhuan He, Darshil Patel, and Aubrey M. Brueckner. Scientific agent skills: A library of procedural knowledge for research agents, 2026. URL https://arxiv.org/abs/2609.00065.

Yan-Shuo Liang and Wu-Jun Li. InfLoRA: Interference-free low-rank adaptation for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23638–23647, 2024. URL https://arxiv.org/abs/2404.00228.

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*, volume 36, pp. 44776–44791. Curran Associates, Inc., 2023. doi: 10.52202/075280-1939. URL https://doi.org/10.52202/075280-1939.

Jorge A. Mendez, Boyu Wang, and Eric Eaton. Lifelong policy gradient learning of factored policies for faster training without forgetting. In *Advances in Neural Information Processing Systems*, volume 33, pp. 14398–14409. Curran Associates, Inc., 2020. URL https://arxiv.org/abs/2007.07011.

NIST/SEMATECH. e-Handbook of Statistical Methods: Full factorial designs. Section 5.3.3.3, n.d. URL https://www.itl.nist.gov/div898/handbook/pri/section3/pri333.htm. Accessed September 26, 2026.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 487–503. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.eacl-main.39. URL https://aclanthology.org/2021.eacl-main.39/.

Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. URL https://arxiv.org/abs/1606.04671.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. URL https://arxiv.org/abs/1707.06347.

Jonathan Schwarz, Wojciech Czarnecki, Jelena Luketina, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & Compress: A scalable framework for continual learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4528–4537. PMLR, 2018. URL https://proceedings.mlr.press/v80/schwarz18a.html.

Mustafa Shukor, Dana Aubakirova, Francesco Capuano, Pepijn Kooijmans, Steven Palma, Adil Zouitine, Michel Aractingi, Caroline Pascal, Martino Russi, Andres Marafioti, Simon Alibert, Matthieu Cord, Thomas Wolf, and Remi Cadene. SmolVLA: A vision-language-action model for affordable and efficient robotics. *arXiv preprint arXiv:2506.01844*, 2025. URL https://arxiv.org/abs/2506.01844.

Davide Tenedini, Riccardo Zamboni, Mirco Mutti, and Marcello Restelli. From parameters to behaviors: Unsupervised compression of the policy space. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=VqnBaeu43F.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109. URL https://doi.org/10.1109/IROS.2012.6386109.

Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuanjing Huang. Orthogonal subspace learning for language model continual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 10658–10671. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-emnlp.715. URL https://aclanthology.org/2023.findings-emnlp.715/.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A. Raffel, and Mohit Bansal. TIES-Merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 7093–7115. Curran Associates, Inc., 2023. doi: 10.52202/075280-0310. URL https://doi.org/10.52202/075280-0310.

Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Sk7KsfW0-.

APPENDIX GUIDE

The appendix separates assumptions and derivations from executed experiments. Appendices A–G give definitions, witness screening, finite-budget diagnostic limits, residual geometry, matched controls, and preservation and complexity arguments. Appendices H–J specify the FTB and VLA instantiations and algorithm outlines. Appendix K then gives exact protocol-specific results, including negative controls and resource accounting. A passing tested arm establishes sufficiency under its stated budget; a failing arm does not establish impossibility for its model class. The later targeted MuJoCo panel and post-outcome Search control are identified separately from the original prospective tracking-12 protocol.

A FORMAL DEFINITIONS AND NOTATION

A.1 POLICY AND TASK

Let

$$\pi_\theta(a \mid s), \qquad \theta \in \Theta \subseteq \mathbb{R}^n$$

denote the policy and its parameters. For a task τ,

$$J_\tau(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{H-1} \gamma^t r_\tau(s_t, a_t)\right]$$

denotes the expected return.

Under finite-rollout evaluation, we estimate the return by

$$\widehat{J}_\tau(\theta).$$

A.2 COMPACT ADAPTATION SPACE

A compact representation

$$G_\phi : \mathcal{Z} \to \Theta$$

induces the reachable policy set

$$\mathcal{S}_\phi = G_\phi(\mathcal{Z}).$$

For a fixed adaptation budget B, let

$$\mathcal{A}_\tau(\mathcal{S}_\phi; B)$$

denote the policy returned after adapting within $\mathcal{S}_\phi$. We define

$$U_\tau(\mathcal{S}_\phi; B) = \mathbb{E}\left[J_\tau\left(\mathcal{A}_\tau(\mathcal{S}_\phi; B)\right)\right].$$

In the main text, we omit B when it is fixed across interventions.

This quantity measures downstream return under a common adaptation budget. Reconstruction error is evaluated separately; a horizon H contains H reward steps.

A.3 Relation to Classical Decompositions

Bottou & Bousquet (2008) distinguish approximation, estimation, and optimization contributions to excess risk. We use this distinction to separate the reachable policy set from finite-budget search within it. Adding and subtracting $V_\tau(\phi)$ gives the population-return identity in Equation (1). Neither this identity nor the independent paired witness evaluation supplies a per-case finite-sample confidence bound.

Ben-David et al. (2010) bound target classification error using source error, a hypothesis-dependent domain divergence, and the combined error of a best joint hypothesis. Our policy-representation analysis follows this structure but uses different quantities. The terms $C_\tau, M_{\mathrm{src}}, K_\tau$ are defined for representations: C_τ compares target and source-average losses, and K_τ is target-only approximation error. They motivate intervention hypotheses without making coverage, mapping, and capacity observable or mutually exclusive causes.

Derivation of the representation bound. Assume Φ_c is nonempty and the quantities in Section 3.1 are finite. The uniform definition of C_τ gives $L_\tau(\phi) \le R_{\mathrm{src}}(\phi) + C_\tau$ and $R_{\mathrm{src}}(\phi) \le L_\tau(\phi) + C_\tau$ for every $\phi \in \Phi_c$. Hence,

$$\begin{aligned} L_\tau(\hat{\phi}) &\le R_{\mathrm{src}}(\hat{\phi}) + C_\tau \\ &= \inf_{\phi \in \Phi_c} R_{\mathrm{src}}(\phi) + M_{\mathrm{src}} + C_\tau \\ &\le \inf_{\phi \in \Phi_c} L_\tau(\phi) + M_{\mathrm{src}} + 2C_\tau \\ &= K_\tau + M_{\mathrm{src}} + 2C_\tau. \end{aligned}$$

The infima need not be attained. Equation (2) therefore follows from our definitions, rather than from a direct application of the classification theorem. A large C_τ indicates possible transfer discrepancy. Testing sufficient repairs still requires interventions.

A.4 Nonidentifiability from a Scalar Performance Observation

Proposition 1 (return-only ambiguity). Suppose the observation is only the final return $Y = J_\tau(G_\phi(\hat{z}))$, with neither the reachable set nor intervened outcomes observed. Even when $J^\star_\tau$ is known, Y does not identify the split (L_τ, O_τ) or whether representation change is necessary.

Proof. Consider a one-step task with two policies θ^+ and θ^- having returns 1 and 0. In one instance the chart contains both policies, but search returns θ^-. It has $(L_\tau, O_\tau) = (0, 1)$ and can be repaired by Search. In another instance the chart contains only θ^-, giving $(L_\tau, O_\tau) = (1, 0)$ and requiring a change to that chart. Both instances have $J^\star_\tau = 1$ and $Y = 0$. Therefore no rule using only these scalar observations can distinguish them. □

This proposition specifies the observation model for the return-only claim. The three-term bound does not uniquely decompose representation loss into information, mapping and capacity causes: it is an upper bound, and its terms depend on the declared class and source distribution. Witness data, chart inspection and intervention outcomes supply additional observations. Our experiments test finite-budget sufficiency under those observations, rather than recovering the population terms $C_\tau, M_{\mathrm{src}}, K_\tau$.

B Behavioral Witness Certification

Starting from a parent policy θ_0, a more flexible task-adaptation operator $\mathcal{F}_\tau$ produces

$$\theta_c = \mathcal{F}_\tau(\theta_0).$$

The generated candidate is checked before it is used as a witness.

Using an independent evaluation set $\mathcal{D}_{\mathrm{eval}}$, we estimate

$$\widehat{\Delta}_c = \widehat{J}_\tau(\theta_c) - \widehat{J}_\tau(\theta_0).$$

A MuJoCo candidate passes the operational witness check if it satisfies the frozen selection screen and its independent paired return difference is positive:

$$\widehat{\Delta}_c > 0.$$

We then set

$$\theta_w := \theta_c.$$

Otherwise, diagnosis abstains. Acceptance uses observed positive gain, not a one-sided confidence bound for every case. Both the tracking-12 and unseen-12 protocols screen eight 512-interaction candidates on eight paired 256-step episodes. The `sign5` rule requires positive mean gain and at least five positive pairs. The selected candidate is evaluated on 64 new paired 256-step episodes. Batch-level advance requires at least 11 positive independent case gains, an exact one-sided sign-test value at most 0.05, and the recorded group-level checks. The separately frozen panels have distinct task IDs and seeds in their manifests.

Training, screening, and independent evaluation use separate samples. VLA Random-20 instead evaluates all 20 frozen target candidates without this per-target gate or abstention. Its directional comparison does not certify 20 independent witnesses.

C Hierarchical Diagnostic Definitions

This section defines the evidence used for abstention and escalation. The natural-task factorial crosses source information with coordinate inversion because their effects may interact. It tests code search, not an isolated decoder update. Capacity is considered when the strongest trained same-capacity candidate fails the frozen acceptance rule. Candidate families are specified before selecting a passing repair.

C.1 Insufficient Evidence

We define

$$\mathcal{H}_{\text{evid}} = \left\{ \widehat{\Delta}_c \leq 0 \right\}.$$

Under $\mathcal{H}_{\text{evid}}$, the observed failure may reflect representation insufficiency, failed optimization, unsuccessful candidate acquisition, or evaluation noise. We therefore make no representation-level intervention:

$$a = \texttt{Abstain}.$$

Abstention leaves representation sufficiency unresolved: the evidence does not justify a change.

C.2 Source Information and Fixed-Capacity Coordinate Inversion

Let D_0 denote the source archive and let D_1 replace a fixed fraction of source samples with independently collected witness state–action evidence. The total number of supervised examples is matched. For each information condition, the decoder is fitted once and held identical across its two coordinate-search arms. Let $\mathcal{A}_0$ use the encoder-provided coordinate and $\mathcal{A}_1$ use the predeclared grid-plus-gradient behavioral inversion on that same fitted decoder. Thus, $\mathcal{A}_1$ optimizes coordinates, not decoder parameters. Define

$$U_{ij} = U_\tau(\mathcal{A}_j(D_i); b)\,.$$

The marginal effects are

$$I^{(j)}_{\text{data}} = U_{1j} - U_{0j}, \qquad I^{(i)}_{\text{inv}} = U_{i1} - U_{i0},$$

and their interaction is

$$I_{\text{int}} = U_{11} - U_{10} - U_{01} + U_{00}.$$

Positive $I^{(j)}_{\text{data}}$ indicates a benefit from target information under inversion program j. Positive $I^{(i)}_{\text{inv}}$ indicates a benefit from stronger inversion under information condition i. A nonzero interaction means that these effects depend on each other. None requires assigning a unique cause from a single archive distance.

A nearest-source behavior distance describes archive coverage but cannot certify a diagnosis. Relevant evidence may already exist in the archive while the current map fails to expose it.

C.3 Diagnostic Schedules and Sequential Repair Paths

The factorial arms are trained separately from a common parent. Their inspection order cannot change the resulting policies. Retaining both fixed-capacity checks prevents escalation whenever either one passes.

Sequential repair is a different question. In the original tracking-12 protocol, U_{11} fits a local patch at the encoded target coordinate and then searches within the patched chart; no reverse path was evaluated in that panel. The later HalfCheetah path intervention compares this operation with searching the old chart first and fitting an equally sized patch at the resulting coordinate. The patch is pinned to vanish at the parent coordinate. Moving away from its narrow support can suppress its learned behavior, so the two operations need not commute. Both paths use the same candidate, archive and target states, 256 patch-fitting updates, 128 coordinate-inversion updates, and 22,528 PPO interactions per arm. A path passes only if it recovers at least 80% of the independently evaluated candidate–parent return gap while keeping archive behavior RMSE at or below 10^{-3}. The preallocated patch is trainable but adds no latent coordinate. The two protocols use different declared fixed-capacity candidate families; their branch counts are not pooled.

C.4 Fixed-Capacity Witness Absorption

Let $b : \Theta \to \mathcal{B}$ denote the common-probe behavior map and let $\mathcal{H}_c$ contain candidates with the fixed architecture, latent dimension, permitted update range, and archive-preservation constraints. Define

$$e_c^\star = \inf_{h \in \mathcal{H}_c} D_B(b(h), b(\theta_w)) \,.$$

Suppose a frozen evaluation set gives empirical error $\widehat{e}_w$, with uniform deviation at most β, and a same-capacity fitting procedure returns h_{fit} whose empirical optimization residual is at most ξ. Then

$$\widehat{e}_w(h_{\text{fit}}) - \xi - \beta \le e_c^\star \le \widehat{e}_w(h_{\text{fit}}) + \beta.$$

The upper bound establishes the existence of a same-capacity witness approximation. The lower bound also needs a uniform estimation bound and a bound on the optimization residual. We lack a global ξ for neural-policy fitting, so a failed fit remains a finite-budget result, not an impossibility proof for the class.

C.5 Nested Candidate Sets

Let $\mathcal{F}_0$ contain frozen-map coordinate searches, $\mathcal{F}_1$ additionally contain witness-conditioned fixed-capacity repairs, $\mathcal{F}_2$ additionally contain the stronger inversion and joint fixed-capacity paths declared for that experiment, and $\mathcal{F}_3$ additionally contain aligned capacity expansions. Retaining all previous candidates gives

$$\mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \mathcal{F}_2 \subseteq \mathcal{F}_3.$$

These sets encode escalation, not the order in which fixed-capacity arms are trained or inspected. The minimum verified level is the smallest k for which a frozen candidate passes both the target-return requirement and archive-preservation criteria. The experiments use frozen finite-budget point-estimate and archive criteria, not simultaneous confidence intervals. A nonpositive independent candidate gain or a zero candidate gap for return-recovery assessment leaves the corresponding diagnosis undetermined; unsuccessful finite-budget arms only justify escalation within the tested family.

C.6 Conditional Behavior-to-Return Bridge

Let μ_t be a probe-state distribution and define the deterministic-policy witness error

$$e_w(\pi;\mu) = \frac{1}{H}\sum_{t=0}^{H-1} \mathbb{E}_{s\sim\mu_t}\left[\|\pi_t(s) - \pi_{w,t}(s)\|_2^2\right].$$

Assume the witness action value is locally L_a-Lipschitz in action and that the repaired policy occupancy satisfies $d_t^\pi \le C_{\mathrm{occ}}\mu_t$. The finite-horizon performance-difference identity and Cauchy–Schwarz give

$$J_\tau(\pi) - J_\tau(\pi_0) \ge \Delta_w - L_a H\sqrt{C_{\mathrm{occ}} e_w(\pi;\mu)}.$$

Under these assumptions, accurate behavior fitting can retain part of the witness gain. Since we do not numerically certify L_a or C_{occ} for neural policies, experiments report both held-out behavior error and closed-loop return.

C.7 Evidence for a Missing Tested Adaptation Direction

Let $\bar{G}$ be the strongest trained same-capacity map that did not pass the frozen target-and-archive rule, and let

$$\theta_b = \bar{G}(z_b).$$

If $\bar{G}$ is differentiable, the local first-order directions reachable by the current compact representation are

$$\mathcal{T}_{\bar{G}}(\theta_b) = \mathrm{col}\left(J_{\bar{G}}(z_b)\right),$$

where

$$J_{\bar{G}}(z_b) = \left.\frac{\partial \bar{G}(z)}{\partial z}\right|_{z=z_b}.$$

For the witness update

$$d_w = \theta_w - \theta_b,$$

we define

$$p_w = P_{\mathcal{T}} d_w$$

and

$$r_\perp = (I - P_{\mathcal{T}})d_w.$$

The relative residual magnitude is

$$\rho_w = \frac{\|r_\perp\|_2}{\|d_w\|_2 + \varepsilon}.$$

The local quantity ρ_w describes geometry. It is neither measured in the FTB experiments nor sufficient to certify a capacity deficit. FTB tests the raw secant d_w from trained U_{11} to the candidate against a random direction of equal norm. The gain Γ_R, evaluated after the fixed-capacity factorial, tests directional value under the stated budget without establishing that the secant lies outside the local tangent.

D GEOMETRY OF RESIDUAL-GUIDED EXPANSION

D.1 FIRST-ORDER UNREACHABLE RESIDUAL

Lemma 1. Assume $\bar{G}$ is differentiable at z_b. For any sufficiently small latent perturbation δz,

$$\bar{G}(z_b + \delta z) - \bar{G}(z_b) = J_{\bar{G}}(z_b)\delta z + o(\|\delta z\|).$$

Therefore, every policy update reachable to first order through the current latent coordinates lies in

$$\mathcal{T}_{\bar{G}}(\theta_b) = \mathrm{col}\left(J_{\bar{G}}(z_b)\right).$$

Since

$$r_\perp = (I - P_{\mathcal{T}})d_w,$$

we have

$$P_{\mathcal{T}} r_\perp = 0$$

and hence

$$r_\perp \perp \mathcal{T}_{\bar{G}}(\theta_b).$$

Thus, no infinitesimal latent update δz can satisfy

$$J_{\bar{G}}(z_b)\delta z = r_\perp.$$

The tangent-orthogonal component is locally unreachable through the current coordinates to first order. A larger move along a nonlinear manifold may still reach a similar policy. The lemma also makes no orthogonality claim about the experimentally added FTB secant d_w.

D.2 WHY RESIDUAL MAGNITUDE IS NOT ENOUGH

A nonzero residual can reflect optimization noise, parameter symmetries, curvature, or behaviorally irrelevant changes. We therefore test an aligned direction against capacity-matched random alternatives, rather than justify expansion from $\|r_\perp\|$ alone.

D.3 BEHAVIOR-SPACE RESIDUAL AND DIRECTIONAL GAIN

Parameter residuals can be behaviorally redundant. For fixed probe states $s_{1:m}$, define

$$\Psi_m(\theta) = \frac{1}{\sqrt{m}}\left[\pi_\theta(s_1), \ldots, \pi_\theta(s_m)\right]$$

and $B(z) = \Psi_m(\bar{G}(z))$ for the strongest trained same-capacity map. At parent coordinate z_b, the local behavior tangent is

$$\mathcal{T}_B = \mathrm{col}(J_B(z_b)), \qquad J_B = J_\Psi(\theta_b)J_{\bar{G}}(z_b).$$

For witness behavior displacement $v = \Psi_m(\theta_w) - \Psi_m(\theta_b)$, let

$$r_B = (I - P_B)v.$$

The least-squares linearized absorption error is

$$\min_{\delta z} \|v - J_B\delta z\|_2^2 = \|r_B\|_2^2.$$

If parameter direction u is added, its new behavior component outside the old tangent is

$$q = (I - P_B)J_\Psi(\theta_b)u.$$

Allowing old coordinates to compensate locally, the reduction in squared linearized residual is

$$\mathrm{Gain}(u) = \frac{(r_B^\top q)^2}{\|q\|_2^2}$$

for $q \neq 0$. Equal-sized directions can differ in how well they align with the remaining behavior residual, motivating the aligned–random comparison. The local probe-space result still requires closed-loop evaluation.

E Matched Controls

E.1 Refit-Null Control

Behavioral repair and refit-null use the same architecture, trainable parameter count, initialization, optimization updates, batch size, optimizer, source archive, parent policy, critic, and continuation budget.

The systematic difference is that

$$R_{\mathrm{beh}}$$

observes witness behavioral supervision, whereas

$$R_{\mathrm{null}}$$

does not.

Therefore,

$$\Gamma_B(\tau) = \widehat{U}_\tau(R_{\mathrm{beh}}; B) - \widehat{U}_\tau(R_{\mathrm{null}}; B)$$

measures the benefit of witness behavior beyond matched refitting. Here $\widehat{U}_\tau$ is the empirical downstream score with a common parent, paired evaluation seeds, normalization and budget B. The prospective MuJoCo score is mean return divided by horizon; VLA uses episode success. The fixed-capacity experiments use this control.

E.2 Spectrum-Matched Random Controls

Suppose a residual matrix has singular value decomposition

$$r_w = U\Sigma V^\top.$$

We construct a random control

$$q = \widetilde{U}\Sigma\widetilde{V}^\top,$$

where $\widetilde{U}$ and $\widetilde{V}$ are randomized orthonormal directions.

This preserves

$$\mathrm{rank}(q) = \mathrm{rank}(r_w),$$

$$\sigma_i(q) = \sigma_i(r_w),$$

and

$$\|q\|_F = \|r_w\|_F.$$

VLA applies this construction separately to each LoRA matrix. The random residual preserves size and spectrum but changes orientation. Define

$$\Gamma_R(\tau) = \widehat{U}_\tau(R_{\text{aligned}}; B) - \widehat{U}_\tau(R_{\text{random}}; B).$$

VLA shares the projection, added rank and continuation protocol; FTB shares the base, added dimension and direction norm. A positive Γ_R indicates useful directional information relative to that random control; neither orthogonality nor expansion necessity follows from this score difference. The pooled 44-case table uses the recorded normalized paired differences, while the prospective subset reports return per horizon step explicitly.

F Preservation of Existing Behaviors

Let the source archive be

$$\mathcal{A} = \{\theta_i\}_{i=1}^N,$$

and denote the corresponding policies after repair by θ_i'.

We define archive-level behavioral drift as

$$D_{\mathcal{A}} = \frac{1}{N} \sum_{i=1}^{N} D_B \left(b(\theta_i'), b(\theta_i) \right).$$

A repair is required to satisfy

$$D_{\mathcal{A}} \leq \epsilon.$$

For the parent policy, an additional anchor constraint may enforce

$$D_B \left(b(\theta_0'), b(\theta_0) \right) \approx 0.$$

For an additive expansion $G_{\text{new}}(z, \alpha) = \bar{G}(z) + \alpha d$,

$$G_{\text{new}}(z_i, 0) = \bar{G}(z_i).$$

This preserves the expansion base, not necessarily the original archive. In particular, relative to original policy θ_i,

$$\pi_{G_{\text{new}}(z_i,0)}(s) - \pi_{\theta_i}(s) = \pi_{\bar{G}(z_i)}(s) - \pi_{\theta_i}(s).$$

Any pre-existing drift remains. The confirmation experiment therefore qualifies the starting checkpoint against the original archived actions and repeats that audit after training with the new coefficient reset. Historical tracking logs and the later reset audit have distinct reference conventions (Tables 20 and 5).

G Representation Complexity

The definition of representation complexity depends on the parameterization.

For an FTB-style latent representation, a simple complexity measure is

$$C(\mathcal{S}) = \dim(\mathcal{Z}),$$

possibly augmented with the parameters required by residual components.

For LoRA-based adaptation,

$$C(\mathcal{S}) = \sum_{\ell} r_\ell \left(d_\ell^{\text{in}} + d_\ell^{\text{out}} \right).$$

For reporting representation changes, define the complexity increase as

$$\Delta C = C(\mathcal{S}_{\phi'}) - C(\mathcal{S}_\phi).$$

Trainable-parameter efficiency is distinct from interaction efficiency, wall-clock efficiency, and total compute. The experimental tables report these quantities separately where measured.

H Instantiation in an FTB-Style Policy Space

In the FTB instantiation,

$$G_\phi(z)$$

is the latent-to-policy decoder.

Fixed-capacity mapping repair modifies

$$\phi \to \phi'$$

while preserving

$$\dim(\mathcal{Z}).$$

In the strict post-mapping MuJoCo test, the aligned direction is $d_w = \theta_w - \theta_b$, where θ_b is the trained U_{11} policy after its matched PPO continuation. The expanded representation is

$$G_{\text{new}}(z, \alpha) = \bar{G}(z) + \alpha d_w.$$

For archived policies, assigning the new coordinate zero ensures

$$G_{\text{new}}(z, 0) = \bar{G}(z),$$

so their decoded policies do not jump when the coordinate is appended. For the target arm, the coefficient is then set to one, which exactly replays the candidate parameters at construction, and is trainable during continuation. The equal-norm random arm undergoes the same coordinate insertion and continuation. This implementation does not calculate the Jacobian projection in Eq. 7.

At z_b, $G_{\text{new}}(z_b, 1) = \theta_w$ is an algebraic identity. The construction tests embedding and using a successful candidate, not whether the old representation lacks a successful policy. Its directional control keeps the added dimension and norm fixed. Replaying the witness, fixing $\alpha = 1$, and searching the scalar without subsequent PPO are not separately evaluated in that protocol. The full parameter direction must also be stored; training one scalar does not imply scalar-only storage.

The later confirmation in Appendix K.3 instead starts from an archive-qualified U_{01}, initializes $\alpha = 0$, fits candidate actions and then runs PPO. It is unconditional with respect to fixed-capacity target success. These initialization and base-policy choices are distinct from the strict post-U_{11} test and historical tracking's linked expansion checkpoints.

The fixed-capacity factorial uses a two-dimensional latent code and four preallocated local patch slots, with a Wendland C^2 compact-support kernel of bandwidth 0.15. Each matched patch receives 256 Adam updates at learning rate 10^{-3}, batch size 16, and 64 behavior states per update. D_0 uses archive evidence, while D_1 replaces half the supervised batch with witness evidence; the

archive pool, initialization, patch center, parent anchor, update count, and optimizer seed are held fixed within each matched pair. A_0 uses the existing encoded target code; A_1 uses direct behavior inversion with a 33-level grid on each latent axis followed by eight gradient starts of 128 steps at learning rate 0.05. The four arms then receive the same 22,528-interaction PPO continuation and are compared on 64 paired decision episodes; the frozen branch decisions are checked on 128 new paired episodes. The stage-one behavior proxy and the final return-recovery decision have separate frozen acceptance rules.

I Instantiation in the LoRA Adaptation Space of a VLA Policy

For the VLA instantiation, let the LoRA updates learned from m source tasks be

$$\Delta_1, \ldots, \Delta_m.$$

They define a source adaptation chart

$$\mathcal{S}_{\mathrm{src}} = \mathrm{span}\{\Delta_1, \ldots, \Delta_m\}.$$

Updates here are tuples of layer matrices with inner product $\langle X, Y \rangle = \sum_\ell \mathrm{tr}(X_\ell^\top Y_\ell)$. For effective LoRA updates, $\Delta_\ell = (a_\ell / r_\ell) B_\ell A_\ell$; the adapter scaling must be included consistently before comparing parameter updates. The builder uses raw $B_\ell A_\ell$, a Gram pseudoinverse with relative cutoff 10^{-8}, and output scaling $a/r = 1$. Its equivalence to effective input updates assumes unit input scaling. The stored geometry audit is a raw-factor reconstruction check, not an independent loaded-model scaling or action-equivalence check.

Given a target witness update

$$\Delta_t,$$

we compute its projection

$$p = P_{\mathcal{S}_{\mathrm{src}}} \Delta_t$$

and residual

$$r = \Delta_t - p.$$

For each residual LoRA matrix,

$$r_\ell = U_\ell \Sigma_\ell V_\ell^\top,$$

we retain the leading k singular directions,

$$r_\ell^{(k)} = U_{\ell,k} \Sigma_{\ell,k} V_{\ell,k}^\top.$$

The aligned adaptation is then constructed as

$$\Delta_{\mathrm{aligned}} = p + r^{(k)}.$$

The source updates define the reusable chart, and the truncated target residual supplies the added direction. Only the untruncated residual is orthogonal under an exact projection. Layer-wise rank truncation can reintroduce a source-space component: for $D = \mathrm{diag}(1, 2)$ and $r = \mathrm{diag}(2, -1)$, $\langle D, r \rangle = 0$, but rank-1 truncation gives $\langle D, r^{(1)} \rangle = 2$. Let $e_k = r - r^{(k)}$. Under the ideal projection,

$$P_{\mathcal{S}_{\mathrm{src}}} r^{(k)} = -P_{\mathcal{S}_{\mathrm{src}}} e_k, \qquad \|P_{\mathcal{S}_{\mathrm{src}}} r^{(k)}\|_F \le \|e_k\|_F.$$

This bounds possible leakage by truncation error; it does not measure leakage in our trained adapters. We consequently describe $r^{(k)}$ as *a truncated source-projected residual*, not a strictly orthogonal new direction. No reprojection after truncation was executed. Reprojection can also increase layer rank, so it would change the tested construction.

I.1 Untruncated Reconstruction and Evaluation Consistency

Without rank truncation,

$$p + r = p + (\Delta_t - p) = \Delta_t.$$

Thus the untruncated Aligned construction and Direct represent the same update in exact arithmetic, provided scaling, non-adapter weights and inference conditions agree. Nominal ranks 136 and 8 describe different factor storage, not different ideal updates. Concatenation, multiplication and serialization can introduce numerical error; rank alone cannot explain different closed-loop outcomes.

The retained builder audits report small raw-update reconstruction errors (Table 25). They do not compare loaded effective weights, same-input actions or repeated identical-model rollouts. The seven Direct–Aligned discordances in Table 24 remain unexplained at that level. We retain the observations as reconstruction/control results, not evidence that the untruncated construction improves the candidate. Truncated residual tests are different interventions and are reported separately.

I.2 Saved Simulator Rollout Examples

Figure 8 shows saved Aligned evaluations from the frozen Random-20 panel. We use the first two tasks in the protocol's task order, LIBERO-90 tasks 48 and 41, and the first logged successful episode for each (zero-based episode indices 1 and 4). Their evaluation records report success. For each video of N frames, we extract frames 0, $\lfloor (N-1)/2 \rfloor$, and $N-1$: indices $(0, 84, 168)$ for task 48 and $(0, 69, 139)$ for task 41. The original 256×256 camera frames are displayed without cropping, flipping, contrast adjustment, or retouching. These selected episodes illustrate the manipulation tasks; aggregate success uses all 200 episodes, independently of this image selection.

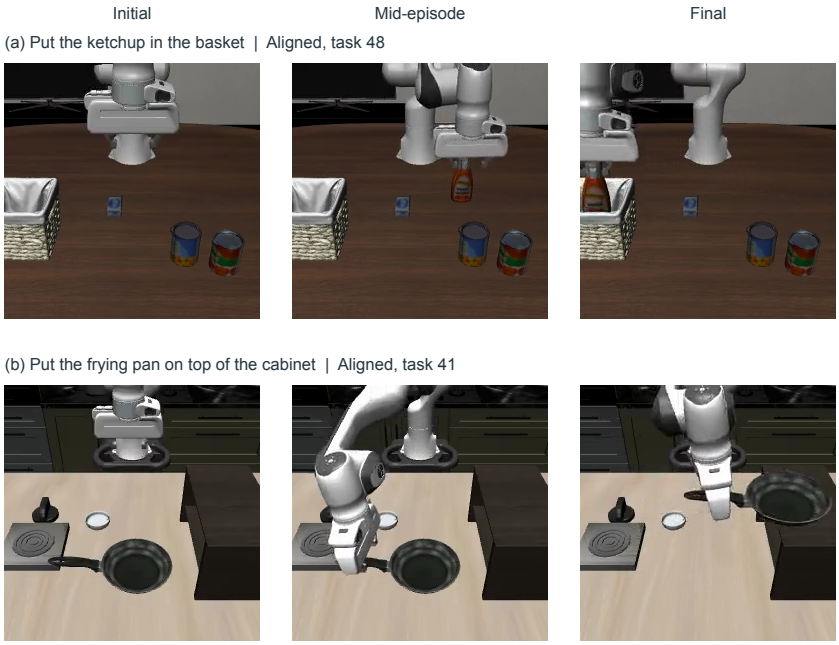

Figure 8: **Actual simulator views from two Aligned evaluation episodes.** Top: lift the ketchup and place it in the basket. Bottom: lift the frying pan and place it on top of the cabinet. Each row shows the initial, middle, and final recorded frame of one episode. The videos, evaluation records, frame indices, and extraction script are retained in the project.

J EXECUTED DIAGNOSTIC PROCEDURES

Procedure 1 (independent MuJoCo witness gate). *Input:* parent θ_0, target τ, frozen candidate-training and evaluation seeds.

1. Train eight candidate policies, each for 512 target interactions.
2. Screen every candidate on eight paired 256-step episodes against the parent. Retain candidates with positive mean gain and at least five positive episode pairs.
3. If none pass the screen, output *abstain*. Otherwise select by the frozen screening rule and evaluate the selected candidate on 64 new paired episodes.
4. If its independent mean gain is nonpositive, output *abstain*; otherwise output witness θ_w.

Procedure 2 (original four-arm fixed-capacity test). *Input:* parent, certified θ_w, archive, and disjoint representation-, mapping-, and evaluation-state partitions.

1. Fit archive-only patch D_0 and, separately, witness-plus-archive patch D_1 under the same patch capacity and optimizer budget.
2. For each fitted decoder, retain the encoded target code A_0 and obtain a stronger inverted code A_1 without changing that decoder's tensors.
3. Form $U_{00} = (D_0, A_0)$, $U_{10} = (D_1, A_0)$, $U_{01} = (D_0, A_1)$, and $U_{11} = (D_1, A_1)$.
4. Give every arm 22,528 PPO interactions, then evaluate paired target return and archive behavior RMSE on frozen seeds.
5. Output the arms satisfying both the 80% candidate–parent return-gap recovery and 10^{-3} archive-RMSE ceiling; otherwise output *fixed-capacity unresolved*.

Procedure 3 (conditional aligned expansion test). *Input:* certified witness and outcomes from the tested nonexpanding arms.

1. If a nonexpanding arm passes, report its weakest tested sufficient level; if incomparable singleton arms pass, report nonuniqueness. Stop without appending a coordinate.
2. Otherwise use the strongest trained same-capacity policy θ_b as the base and construct the FTB secant $d_w = \theta_w - \theta_b$.
3. Append either the aligned d_w or an equal-norm random direction; train both with matched PPO continuation budgets and evaluate target return and archive behavior on frozen seeds.
4. Report a passing intervention, or *unresolved* if none pass. Report candidate-generation, diagnosis, persistent-storage, and continuation costs separately.

The VLA source-chart projection and truncated residual in Appendix I are a separate directional test; VLA does not execute Procedures 1–3 as a complete hierarchical procedure.

K EXPERIMENTAL PROTOCOLS, EXACT RESULTS, AND AUDIT BOUNDARIES

This appendix records the MuJoCo mechanism and repair tests, followed by VLA directionality, robustness, and continuation costs. Each panel is reported separately. A chart–target–seed case is not an independent task, and a 128-episode reevaluation checks a saved checkpoint rather than sampling new tasks. Offline decision replay also differs from executing an expansion. The package's `figures/source_data/` directory records additional source locations and result scope. Reproducing training requires the experiment code and checkpoints, which are maintained separately.

K.1 MUJOCO: CONTROLLED DIAGNOSTIC TESTS

Table 1: Controlled constructions, reported separately from natural tasks. Linear target error is squared behavioral error. Rotations and local constructions are controlled test units, not independent task draws.

Test	Size or setting	Observed result
Linear Search	8 dimensions, 2 archive behaviors	Error $0.64 \to 0$
Linear Mapping	Same fixed-capacity construction	Error $0.25 \to 0$
Rotated planted levels	1,000 orthogonal rotations	All three levels identified
Local trained-FTB constructions	800 across three environments	All three levels identified
Held-out nonlinear Coverage	96 cases	96/96 correctly selected
Held-out nonlinear Mapping	96 cases	96/96 correctly selected
Held-out nonlinear Capacity	96 cases	96/96 correctly selected

The trained-chart constructions use HalfCheetah, Walker2d and Hopper. Held-out outcome replay is tabulated separately in Table 14; replay reuses those outcomes and does not execute a new repair.

K.2 MUJOCO: MATCHED MECHANISM CONTROLS

Table 2: Matched MuJoCo mechanism comparisons over 44 frozen full-budget cases. The first row changes only access to witness behavior at fixed repair capacity; the second row changes only the orientation of the added residual at matched nominal size and norm.

Comparison	Wins/Ties/Losses	Mean normalized advantage
Witness repair vs. matched refit	42/1/1	$+0.045717$
Aligned residual vs. matched random	43/1/0	$+0.047314$

Here "full-budget" refers to the pre-specified optimization budget, not full-parameter training. The refit comparison tests behavioral information; the random comparison tests task alignment.

Table 3: Target cost on the complete 12-case prospective HalfCheetah subset. Cost is negative return per horizon step; lower is better. Means are computed from the packaged `mechanism_prospective12.json`. The two comparisons are separate interventions, not training stages.

Comparison	Arm	Mean target cost
Behavioral information	Matched archive refit	1.036528
	Witness repair	0.984031
Direction alignment	Matched random	1.020946
	Aligned coordinate	0.968573

K.3 MUJOCO: NEW-TASK ARCHIVE AND DECODER-UPDATE CONFIRMATION

We draw two rewards with seed 260929 from a speed–scale grid, excluding speeds already in the task registry. The resulting pairs are $(1.15, 0.75)$ and $(2.15, 0.60)$. Three source charts and two

seeds per chart give 12 cases. Both confirmation experiments use this same development-disjoint panel.

Capacity starts from the full-budget U_{01} checkpoint. Qualification requires archive action RMSE at most 10^{-3}, regardless of target recovery. Eight cases qualify; all four unqualified cases remain listed. This is an unconditional aligned-versus-random mechanism test, not an execution of the hierarchical stopping rule. Archive qualification does not require prior Information or decoder-repair failure. Four qualified cases also pass a fixed-capacity arm in the separate 64-episode factorial: seeds 713002, 713102, 714011 and 714211 (Table 8). Capacity uses a different base and 128-episode references; the panels are not pooled as one nested success rate. In particular, $8/8$ does not mean eight necessary expansions. The new coordinate is a scalar multiplying the full candidate-minus-U_{01} parameter direction, or one of three equal-norm random directions. It is not a rank-1 matrix approximation. The old code, decoder, and fitted patch stay frozen. All four arms start at $\alpha = 0$ with identical weights and share 512 candidate-action fit states, 512 disjoint check states, 256 Adam coefficient updates at learning rate 0.05, minibatch 128, gradient clipping 10 and coefficient bound $|\alpha| \leq 2$. They then receive 22,528 PPO interactions at coefficient learning rate 5×10^{-5} and critic learning rate 10^{-3}, with four environments, 128-step rollouts, four update epochs and minibatch 128. Final evaluation uses 128 paired episodes of horizon 256, separate from fitting and tuning. A pass requires a positive candidate–parent gap, at least 80% recovery and archive RMSE at most 10^{-3}.

Table 4: Capacity confirmation on the eight archive-qualified cases. Any random counts cases with at least one passing direction, not a pool of 24 runs. Four of the full 12 cases lack a qualified base. The U_{01} checkpoint is a reference without additional Capacity-stage PPO.

Tested policy	Target-and-archive passes
Fixed-capacity U_{01} checkpoint	0/8
Aligned new coordinate	**8/8**
Random coordinate 0	1/8
Random coordinate 1	0/8
Random coordinate 2	0/8
Any of the three random coordinates	1/8

After PPO, we reset the added coefficient to zero for each archived policy. The old chart state is unchanged. Table 5 records the audit size, threshold and observed maximum, checking recovery of old-code behavior rather than old-task return with the target coefficient active. Although the coefficient starts at zero, common candidate-action fitting writes it before PPO. Costs therefore include the full-weight direction and candidate generation. At the target base coordinate, setting $\alpha = 1$ reproduces the candidate algebraically. Retained final Aligned coefficients lie between 0.998812 and 1.012137, close to that endpoint. The test establishes candidate-direction utility under matched write-in and PPO, with audited old-code recovery. A fixed-$\alpha = 1$ control and scalar-search-only control were not separately evaluated, so the result does not isolate the benefit of PPO over replaying the witness.

The two-level factorial retains all four combinations to distinguish single from joint sufficiency (NIST/SEMATECH, n.d.). Information changes supervised patch initialization: I_0 fits archive behavior and I_1 fits witness-conditioned behavior. It does not add data while holding initialization constant. During PPO, M_0 freezes the fitted decoder, whereas M_1 updates the preallocated patch as well as the two coordinates. Neither adds a patch slot or latent coordinate.

All four arms use 22,528 PPO interactions. The decoder-update arms use learning rate 10^{-4} for code and patch, patch-change norm at most 0.003, and archive penalty 30; these settings were frozen using the separate development panel. The archive constraint uses eight old policy codes and even-index states; disjoint odd-index states are reserved for the final archive audit. Final target evaluation uses 64 paired episodes of horizon 256, with the same seeds across all four cells. The final return threshold is the frozen parent return plus 80% of the candidate–parent gap. Both the target and the 10^{-3} archive ceiling must pass.

The joint arm has training archive RMSE 2.66×10^{-5} and disjoint-audit RMSE 2.68×10^{-5}. Exactly one of 12 cases meets the predeclared joint-only rule: I_0M_0, I_1M_0, and I_0M_1 fail, while

Table 5: Capacity qualification, training budget and reset audit. The audit includes candidate-action write-in and PPO, with the added coefficient reset to zero after training. It tests archived-state behavior.

Item	Value
Full confirmation panel	12 cases
Archive-qualified / unqualified starting checkpoints	8 / 4
Archive RMSE ceiling	0.001
Initial added coefficient	$\alpha = 0$
Trainable policy parameters per Capacity arm	1 scalar
Candidate-action fitting / check states	512 / 512
Coefficient fitting updates	256
PPO interactions per trained arm	22,528
Final evaluation episodes per arm	128
Archived policies / audit states	32 / 188
Maximum reset-audit RMSE	0.0007904

Table 6: Executed decoder-factorial configurations and complete panel results. Policy-side trainable counts refer to downstream PPO, excluding the shared critic and earlier supervised initialization.

Arm	Patch initialization	Decoder in PPO	Trainable policy	Passes
I_0M_0	Archive-only	Frozen	2	0/12
I_1M_0	Witness-conditioned	Frozen	2	**3/12**
I_0M_1	Archive-only	Local patch updated	1,832	0/12
I_1M_1	Witness-conditioned	Local patch updated	1,832	**3/12**

Table 7: The joint-only case on reward $(2.15, 0.60)$, chart c, seed 714211. Recovery is measured relative to the parent–candidate return gap. Frozen reference returns per step are -0.567660 and -0.560411; the target threshold is -0.561861. All four cells satisfy the archive ceiling. This identified example is not a prevalence estimate.

Intervention	Return per step	Gap recovery	Pass
No repair (I_0M_0)	-0.566682	13.5%	No
Information only (I_1M_0)	-0.565682	27.3%	No
Decoder update only (I_0M_1)	-0.567111	7.6%	No
Joint repair (I_1M_1)	$\mathbf{-0.557030}$	**146.6%**	Yes

I_1M_1 passes. Information and joint repair each resolve three cases overall. The joint-only example therefore establishes existence, not an average interaction advantage.

Table 8: Every case in the new confirmation panel. All decoder cells use the target-plus-archive criterion. Capacity N/A denotes the absence of an archive-qualified U_{01} starting checkpoint.

Case	I_0M_0	I_1M_0	I_0M_1	I_1M_1	Aligned Capacity
`confirm2_t115_a_s713001`	Fail	Fail	Fail	Fail	N/A
`confirm2_t115_a_s713002`	Fail	Pass	Fail	Fail	Pass
`confirm2_t115_b_s713101`	Fail	Fail	Fail	Fail	Pass
`confirm2_t115_b_s713102`	Fail	Pass	Fail	Pass	Pass
`confirm2_t115_c_s713201`	Fail	Fail	Fail	Fail	N/A
`confirm2_t115_c_s713202`	Fail	Fail	Fail	Fail	Pass
`confirm2_t215_a_s714011`	Fail	Pass	Fail	Pass	Pass
`confirm2_t215_a_s714012`	Fail	Fail	Fail	Fail	Pass
`confirm2_t215_b_s714111`	Fail	Fail	Fail	Fail	Pass
`confirm2_t215_b_s714112`	Fail	Fail	Fail	Fail	N/A
`confirm2_t215_c_s714211`	Fail	Fail	Fail	Pass	Pass
`confirm2_t215_c_s714212`	Fail	Fail	Fail	Fail	N/A

Portable figure inputs retain terminal means, all case gates, archive audit values and the frozen configurations in `figures/source_data/confirmation_results.json`. The original stage folders are `capacity_retry_v1`, `decoder_mixed` and `full_budget` within the `run_archive_mixed_confirmation2_v1` experiment.

K.4 MuJoCo: case-level single and joint repair examples

Figure 4 compares five repairs using four chart–seed identifiers. For each sufficient-intervention category, we display the lexicographically last eligible identifier. Panel e uses the sole joint-only case. This display rule was chosen after observing outcomes, not preregistered as a new evaluation. The JSON in `figures/repair_case_examples/source_data.json` retains all cohort decisions and displayed measurements. Its companion CSV records each arm's return, reference, archive audit, episode count, and policy-side trainable parameter count.

Each bar is the mean return over paired evaluation episodes, normalized as $100(J_{\mathrm{arm}} - J_{\mathrm{parent}})/(J_{\mathrm{candidate}} - J_{\mathrm{parent}})$. The target criterion remains 80%, with archive ceiling 10^{-3}. All shown arms satisfy their available archive audits, so return recovery determines whether they pass. Blue marks the focal intervention, not every passing arm. Scores retain negative recovery and values above 100%, which mean exceeding the candidate gain. Bars are sorted by score within each panel. These selected examples show sufficiency, not population success rates or unique causes.

Table 9: Case identities and distinct protocols for Figure 4. Target columns give the HalfCheetah reward's speed and scale. Evaluation episodes are per arm, not independent target tasks.

Panel	Case identifier	Target	Cohort size	Episodes
a	`extended172_c_s706211`	$(1.72, 0.78)$	12	64
b	`extended172_b_s706111`	$(1.72, 0.78)$	12	64
c	`gap210_b_s707122`	$(2.10, 0.75)$	24	64
d	`confirm2_t215_c_s714211`	$(2.15, 0.60)$	12	128
e	`confirm2_t215_c_s714211`	$(2.15, 0.60)$	12	64

Table 10: Every displayed arm in Figure 4, in ascending score order within each panel. Recovery is a percentage of the candidate–parent return gap, not success probability. All arms satisfy the archive gate; Table 9 identifies cases and episode counts. Raw returns and archive audits remain in the companion CSV.

Panel	Intervention	Gap recovery (%)	Target-and-archive pass
a	No repair	−0.9	No
	Stronger search	80.3	Yes
b	No repair	−0.6	No
	Stronger search	1.6	No
	Code inversion	4.5	No
	Information repair	96.9	Yes
c	No repair	−0.2	No
	Information only	30.1	No
	Joint repair	105.7	Yes
	Decoder only	130.5	Yes
d	Random 1	9.6	No
	Random 2	12.0	No
	Fixed base	32.2	No
	Random 3	35.3	No
	Aligned coordinate	99.6	Yes
e	Decoder only	7.6	No
	No repair	13.5	No
	Information only	27.3	No
	Joint repair	146.6	Yes

Panels a,b come from targeted natural-12. Panel a compares ordinary continuation with stronger search on the unchanged decoder. This search uses no candidate for training or selection. Panel b compares continuation, stronger search, inversion, and witness-conditioned patch fitting at fixed coordinate dimension. Information repair fits the patch under supervision, then runs PPO with the decoder frozen.

Panel c is drawn from a separate decoder-24 factorial. Its full-cohort target-and-archive pass counts, in no-repair, Information-only, decoder-only, and joint order, are $0/24, 4/24, 8/24, 7/24$. The selected decoder-only arm recovers 130.5% of candidate gain, while the joint arm also passes at 105.7%; both remain visible. Decoder updating trains the preallocated 1,830-parameter patch and two coordinates during 22,528 PPO interactions. Frozen-decoder arms train two coordinates with the same interaction count. These arms match interaction budgets, not trainable parameter counts. The complete four-cell decisions are retained in the accompanying JSON rather than inferred from the selected example.

Panels d,e use the development-disjoint confirmation in Appendix K.3. They share a chart–seed identifier but use different starting policies, references, and evaluation protocols, rather than successive steps of one trajectory. Capacity tests each of three random directions against aligned. Every coefficient starts at zero, receives common candidate-action write-in, and then PPO. The base has no extra Capacity-stage PPO, so aligned versus random is the matched contrast. In the decoder factorial, both single interventions fail in the displayed joint-only example. Full-cohort counts remain $0/12, 3/12, 0/12, 3/12$.

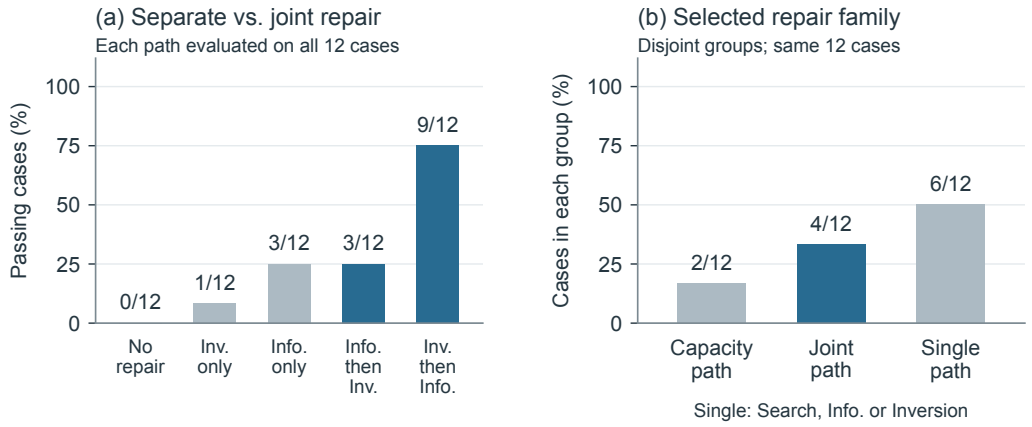

Figure 9: **Complete targeted natural-task cohort.** This cohort-level view complements the selected case-level examples. (a) Passing counts for independent paths over all 12 cases; memberships can overlap. (b) Disjoint weakest-passing groups: six single-path, four joint-path, and two Capacity cases. These joint paths combine fitting with code inversion, not decoder parameter updates. The unchanged cumulative union is $0/12, 3/12, 5/12, 6/12, 10/12, 12/12$.

K.5 MuJoCo: natural-task diagnosis and repair

Table 11: Original prospective tracking panel: repair selection and cost. Interaction counts cover follow-up diagnosis and repair, not candidate generation. The later Search control is reported separately.

Rule	Target-and-archive passes	Expansions	Interactions
Hierarchical selection	12/12	11	1,329,152
Always-expand	12/12	12	270,336

In the original prospective tracking panel, each fixed-capacity arm receives 22,528 PPO interactions. Stronger inversion on the archive-refitted decoder resolves one of 12 cases with 98.4% candidate-gap recovery. The remaining 11 proceed to aligned expansion. All branch decisions persist on 128 fresh paired episodes. Both the hierarchy and always-expand pass 12/12, using 1,329,152 and 270,336 follow-up interactions, respectively. The hierarchy thus avoids one persistent expansion at additional diagnostic cost. A later control resolves the same case using the untouched decoder, establishing Search sufficiency.

On previously unused HalfCheetah rewards, all tested fixed-capacity arms fail in 11 cases with positive independent candidate gains. Starting from the trained witness-fitting-plus-inversion policy (U_{11}), the raw candidate-minus-base direction passes 11/11, compared with 1/11 for an equal-norm random direction and 0/11 for the starting policy. This unprojected-secant protocol differs from the zero-initialized coefficient tests below and the archive-preserving confirmation in Appendix K.3.

Table 12: Executed configuration of the original natural-task factorial. The archive-only patch fit is shared by U_{00}/U_{01}; the witness-conditioned patch fit is shared by U_{10}/U_{11}. Within each pair the decoder tensors are identical before PPO, so the U_{01}–U_{00} contrast isolates coordinate inversion, not decoder remapping.

Arm	Fitted patch	Target code	Downstream PPO
U_{00}	Archive-only	Existing encoder	22,528 interactions
U_{10}	Witness + archive	Existing encoder	22,528 interactions
U_{01}	Archive-only	Behavior inversion	22,528 interactions
U_{11}	Witness + archive	Behavior inversion	22,528 interactions

Each arm has the same two-dimensional latent space and four preallocated Wendland-C^2 patch slots at bandwidth 0.15; only the allocated patch vector is fitted while the base decoder is frozen. Patch fitting minimizes mean-squared action difference on sampled states for 256 Adam updates at learning rate 10^{-3}, batch size 16, and 64 sampled behavior states per update. Archive-only fitting uses source-policy actions; the witness fit replaces half of each policy minibatch with candidate actions. The inversion arms use a 33×33 latent grid and eight 128-step Adam starts at learning rate 0.05 on 128 disjoint candidate mapping states, updating the latent code only. The 128 representation-fit states, 128 mapping-fit states, and 256 held-out candidate states are disjoint. The fitted arms then receive the same 22,528-interaction PPO continuation and paired evaluation seeds. Thus the natural U_{01} success is a finite-budget Search/Inversion success.

Table 13: Fixed-coordinate witness-patch repair on two held-out HalfCheetah reward settings. Three source charts and two seeds per chart give 12 chart–target–seed cases, not 12 independent tasks. Each arm receives 22,528 PPO interactions and must recover at least 80% of the candidate–parent return gap with archive behavior RMSE at most 10^{-3}. The trainable patch was preallocated; bandwidth 0.002 was selected on development settings. The 128-episode column reevaluates the same checkpoints.

Fixed-capacity arm	64 episodes	128 fresh episodes
U_{00}: archive refit, original mapping	0/12	0/12
U_{01}: archive refit, stronger code inversion	0/12	0/12
U_{10}: witness patch, original mapping	3/12	3/12
U_{11}: witness patch, stronger code inversion	3/12	3/12

All 12 candidates pass the independent witness gate. The three U_{10} selections and all 12 branch decisions persist on new episodes. Witness fitting updates a preallocated patch while keeping the target coordinate fixed. With the mapping fallback, U_{11} keeps the same code and matches U_{10}. This panel is separate from tracking-12, where the single fixed-capacity resolution came from stronger inversion on an archive-refitted decoder. That original comparison alone did not isolate a decoder update or establish untouched-decoder sufficiency.

The 288 controlled constructions share source charts. Search-only cases belong to a separate analytic control. This replay measures branch choice, not the training cost or persistent growth of executed expansions. The planted cases isolate individual limitations, so they do not test mixed failures or

Table 14: Offline decision-rule ablation on frozen outcomes of 288 controlled held-out nonlinear FTB cases: 96 planted cases each for Coverage, Mapping, and Capacity. Retaining both fixed-capacity checks avoids premature escalation in these planted cases, regardless of their inspection order. A premature Capacity test is a replayed decision, not an executed expansion.

Decision rule	Correct branch	Premature Capacity test
Information → Mapping	288/288	0/288
Mapping → Information	288/288	0/288
Skip Information	192/288	96/288
Skip Mapping	192/288	96/288
Capacity first	96/288	192/288

establish that Information must precede Mapping. Natural-task results report the weakest sufficient *tested* route under each experiment's frozen budget.

In the separate planted-Capacity check formerly plotted in Figure 2, relative held-out target errors after the strongest tested fixed-capacity intervention are 0.909 for HalfCheetah, 0.813 for Walker2d, and 0.411 for Hopper, all above the frozen acceptance threshold of 0.1. These error measurements are not pooled with the 288 identification decisions.

Table 15: Sequential fixed-capacity repair paths on two other held-out HalfCheetah reward settings. The paths share candidate, patch, inversion, and PPO budgets. The 128-episode check reevaluates the same frozen checkpoints; it is not another target-task sample.

Panel	Info. → Invert.	Invert. → Info.
New targets, 64 paired episodes	0/12	3/12
Same checkpoints, 128 new episodes	0/12	3/12

Inversion followed by Information resolves three cases that the reverse order does not. One also passes stronger inversion alone, so adding the reverse path resolves two cases beyond the original four-arm family. All 12 path decisions persist on fresh rollouts. An exploratory bandwidth-0.01 sweep gives $3/12$ versus $7/12$ for the two paths on previously seen development settings. That sweep is development evidence, separate from the held-out confirmation.

Table 16: First sufficient path on a targeted two-reward HalfCheetah panel. Twelve chart–target–seed cases are not twelve independent task draws. Both target return and fixed-code archive RMSE must pass. Rewards and budgets were frozen before candidate outcomes, but the rewards were selected after prior development inspection. Inversion is code search on an archive-refitted decoder, not an isolated decoder update.

First sufficient tested path	Cases
Pure Search on unchanged representation	3/12
Witness Information at the existing code	2/12
Archive refit plus stronger code inversion	1/12
Code inversion then witness Information	4/12
Aligned expansion after tested fixed-capacity paths	2/12
Positive independent candidates; cases with a passing path	12/12; 12/12

The cumulative target-and-fixed-code-archive acceptance is $3 \to 5 \to 6 \to 10 \to 12$ under this route. Across eight capacity-eligible cases, aligned expansion passes the target threshold in $8/8$ and matched random in $1/8$; with the fixed-code archive criterion, the counts are $7/8$ and $1/8$. The aligned archive failure remains in the record. An earlier 12-case panel revisited with stronger same-capacity mapping search after its outcomes were known gives $2/2/1/3/3$ Search/Information/inversion/reverse-joint/Capacity cases and one nonpositive candidate; this is development sensitivity, not independent confirmation.

F01 resolves the pre-identified U01 case, while the other 11 cases remain unresolved. Its decoder tensors are unchanged and archive behavior RMSE is zero. The saved F01 checkpoint also passes

Table 17: Post-outcome disambiguation on all 12 original tracking cases. F00 and F01 use the untouched decoder; U00 and U01 reuse the archived archive-refitted decoder. Within each decoder condition only the target-code search differs. New F arms use the original candidate states, inversion settings, 22,528-interaction PPO continuation, 64 paired evaluation episodes, 80% candidate-gap threshold, and archive RMSE ceiling 0.001. The untouched arms intentionally fit fewer parameters before PPO.

Decoder condition	Existing code	Strong inversion
Untouched FTB decoder	F00: 0/12	F01: 1/12
Archive-only refitted decoder	U00: 0/12	U01: 1/12

the recomputed candidate-gap threshold on 128 fresh paired episodes. This checks stability and Search sufficiency for the same case. It neither adds a new target task nor shows that the other cases require decoder changes.

K.6 MuJoCo: coefficient learning and candidate behavior write-in

Table 18: MuJoCo coefficient-learning controls from the same trained U_{11} starting policy. Zero-only PPO and behavior write-in are different protocols and are not pooled. The confirmation cohort contains six cases, five with certified witnesses and one retained abstention. Each capacity arm then receives 22,528 PPO interactions.

Protocol	Aligned joint passes	Matched-random passes
Ordinary PPO from $\alpha = 0$, held-out	0/11	0/11 (each direction)
Behavior write-in from $\alpha = 0$, confirmation	5/5	0/5 (each direction)

Write-in fits only the new coefficient using 512 candidate-action states, 512 disjoint check states, and 256 Adam updates at learning rate 0.05. Aligned and the three equal-norm random controls share states, targets, minibatch order, and continuation budget. The ordinary zero-initialized PPO failure remains in the results. Write-in succeeds with candidate supervision, not through candidate-free learning.

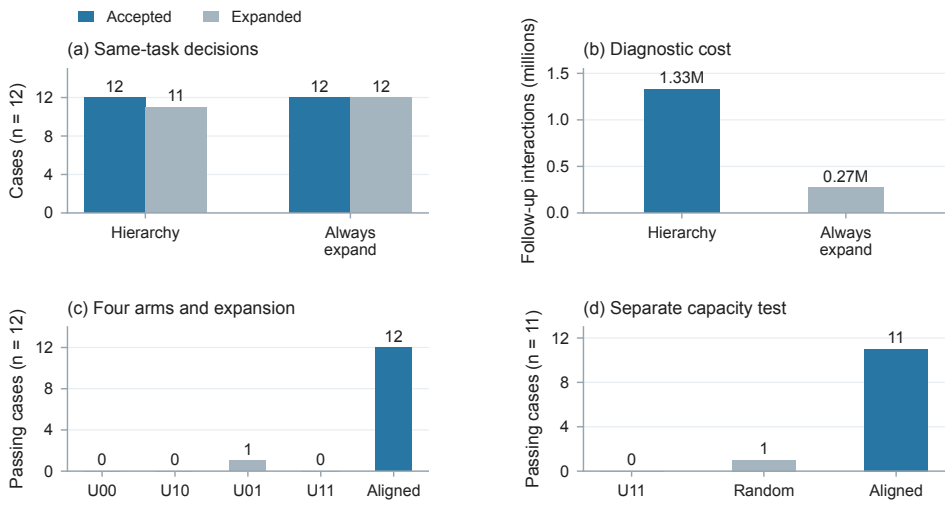

Figure 10: **Original prospective tracking-12 protocol, retained separately.** The four-arm protocol reaches one fixed-capacity resolution relative to archive refitting and 11 linked aligned expansions. Historical "mapping-only" labels denote code inversion, not decoder updates. The later all-case control shows that the sole fixed-capacity case also resolves by stronger Search on the untouched decoder. Interaction totals exclude candidate generation. Panel (d) uses distinct held-out Capacity cases, not tracking-12.

K.7 CASE-LEVEL RETURN AND DECISION AUDIT

Table 19: Original prospective tracking-12: closed-loop mean return per horizon step on 64 paired evaluation episodes. Larger is better. The suffix gives target speed (075 or 175), source chart (a–c), and run (1–2). Only U_{01} in case 075-a-1 passes both the 80% candidate–parent return-gap rule and archive RMSE ceiling 10^{-3}; a high return alone is not sufficient. This table uses the original four-arm protocol, not the later targeted panel.

Case	Parent	Candidate	U_{00}	U_{10}	U_{01}	U_{11}
075-a-1	-1.7582	-1.7063	-1.7481	-1.7428	-1.7071†	-1.6986
075-a-2	-1.7573	-1.6355	-1.7478	-1.7467	-1.7315	-1.7433
075-b-1	-1.3360	-1.2573	-1.3216	-1.3251	-1.3268	-1.3240
075-b-2	-1.3274	-1.2362	-1.3187	-1.3183	-1.3135	-1.3163
075-c-1	-1.5508	-1.4867	-1.5187	-1.5334	-1.5301	-1.5428
075-c-2	-1.5493	-1.4911	-1.5435	-1.5379	-1.5259	-1.5263
175-a-1	-0.6106	-0.5894	-0.6110	-0.6094	-0.6031	-0.6006
175-a-2	-0.6085	-0.6005	-0.6085	-0.6105	-0.6046	-0.6077
175-b-1	-0.4514	-0.4313	-0.4496	-0.4492	-0.4467	-0.4464
175-b-2	-0.4496	-0.4286	-0.4497	-0.4515	-0.4527	-0.4514
175-c-1	-0.5013	-0.4926	-0.5014	-0.4980	-0.5093	-0.5083
175-c-2	-0.4970	-0.4861	-0.4959	-0.4967	-0.4949	-0.4949

Table 20: Original tracking-12: archive RMSE of the two fixed-capacity decoders and final 128-episode verification. U_{00}/U_{01} share D_0; U_{10}/U_{11} share D_1. Selected returns and thresholds are copied from the final logs; reference means are given in Table 21. Logs assign zero archive RMSE to selected Aligned arms, but do not identify their reference decoder. U_{01} is selected only for 075-a-1.

Case	RMSE D_0	RMSE D_1	Return threshold	Selected return	Selected arm
075-a-1	0.000119	0.001516	-1.7097	-1.7064	U_{01}
075-a-2	0.000171	0.000546	-1.6383	-1.6013	Aligned
075-b-1	0.000252	0.000343	-1.2713	-1.2283	Aligned
075-b-2	0.000293	0.000932	-1.2515	-1.1997	Aligned
075-c-1	0.000158	0.000493	-1.5060	-1.4837	Aligned
075-c-2	0.000200	0.000280	-1.5105	-1.4827	Aligned
175-a-1	0.000200	0.000507	-0.5922	-0.5846	Aligned
175-a-2	0.000174	0.001770	-0.6010	-0.5980	Aligned
175-b-1	0.000328	0.000928	-0.4358	-0.4297	Aligned
175-b-2	0.000261	0.001572	-0.4333	-0.4248	Aligned
175-c-1	0.000173	0.000896	-0.4947	-0.4892	Aligned
175-c-2	0.000278	0.000962	-0.4899	-0.4856	Aligned

The first table uses `information_mapping_full_budget` records; the second joins their decoder audits to `frozen_final_rollout` records by case ID. Expansion checkpoints are linked from the earlier full-budget comparison, not recomputed from each final U_{11} decoder. The zero values are retained logged readouts; they do not certify removal of the nonzero D_1 drift. The saved tracking records do not specify the expansion base/reference pair or a rollback audit, so these zero values are not used as evidence of original-archive recovery. The later confirmation performs that audit explicitly.

Table 21: Reference means used for the original tracking-12 final decision, all evaluated on the same 128 paired episodes as the selected policies. Return is per horizon step. Each recorded threshold is exactly $u_{\mathrm{req}} = \widehat{J}_{\mathrm{parent}} + 0.8(\widehat{J}_{\mathrm{candidate}} - \widehat{J}_{\mathrm{parent}})$ before rounding. These references differ from the 64-episode decision means in Table 19.

Case	Parent (128)	Candidate (128)	u_{req} (128)
075-a-1	-1.76079570	-1.69686832	-1.70965380
075-a-2	-1.75353637	-1.60948695	-1.63829683
075-b-1	-1.33282577	-1.25589361	-1.27128004
075-b-2	-1.32417739	-1.23330883	-1.25148254
075-c-1	-1.55139742	-1.49470569	-1.50604404
075-c-2	-1.55019645	-1.50063155	-1.51054453
175-a-1	-0.60696814	-0.58846001	-0.59216163
175-a-2	-0.61033082	-0.59867128	-0.60100318
175-b-1	-0.45167515	-0.43183276	-0.43580124
175-b-2	-0.44965964	-0.42915510	-0.43325601
175-c-1	-0.50168483	-0.49291436	-0.49466846
175-c-2	-0.49777215	-0.48788954	-0.48986606

The fresh reevaluation uses saved checkpoints, not new target tasks. Point-estimate branch decisions have no per-case confidence guarantee; cases near the threshold may change under additional sampling. The packaged `appendix_consistency.json` retains unrounded references, paired returns and seeds for all 12 cases so the thresholds can be recomputed independently.

Table 22: Later targeted two-reward panel: all 12 chart–target–seed decisions. Each entry indicates whether target and archive criteria both pass. Columns are unchanged-representation Search (S), witness Information (I), archive-refit plus code inversion (V), Information then inversion (F), inversion then Information (R), and aligned expansion (C). Code inversion is Search, not a decoder-weight update. The first sufficient path follows the declared route; later paths may also pass.

Case	S	I	V	F	R	C	First sufficient
074-a-1	–	–	–	–	–	Yes	Capacity
074-a-2	–	–	–	–	–	Yes	Capacity
074-b-1	–	Yes	–	Yes	Yes	–	Information
074-b-2	–	–	–	–	Yes	–	Reverse joint
074-c-1	Yes	–	–	–	Yes	Yes	Search
074-c-2	–	–	–	–	Yes	Yes	Reverse joint
172-a-1	–	–	Yes	–	Yes	–	Refit + inversion
172-a-2	Yes	–	–	–	Yes	Yes	Search
172-b-1	–	Yes	–	Yes	Yes	–	Information
172-b-2	–	–	–	–	Yes	Yes	Reverse joint
172-c-1	Yes	Yes	–	Yes	–	–	Search
172-c-2	–	–	–	–	Yes	Yes	Reverse joint

The targeted decisions are retained in the packaged `figures/source_data/natural_cases.json`; its legacy key `mapping` denotes the archive-refit-plus-inversion route, not decoder-weight training. Search, Information, refit-plus-inversion and joint paths are not mutually exclusive causal classes: the table reports the weakest passing *tested route*. No mixed-failure identifiability claim follows from these case counts.

K.8 VLA: RANDOM-20 DIRECTIONAL TRANSFER

Table 23: Random-task VLA evaluation on 20 randomly sampled LIBERO-90 targets. Each arm has 200 closed-loop episodes (10 per target). The three random arms match the aligned residual in layer-wise rank and singular-value spectrum while randomizing its directions.

Arm	Successes	Rate	Task-level readout
Direct rank-8 candidate	79/200	39.5%	Absolute reference
Aligned rank-8 residual	69/200	34.5%	17/20 above best random
Matched random, seed 0	0/200	0.0%	Matched control
Matched random, seed 1	0/200	0.0%	Matched control
Matched random, seed 2	0/200	0.0%	Matched control

The ratio $69/79 = 87.3\%$ compares pooled success counts. It does not measure success on the same individual episodes. The result applies to the frozen SmolVLA/LIBERO-90 protocol, without testing transfer across embodiments or VLA families.

K.9 VLA: HELD-OUT SUITES AND SOURCE-QUALIFIED CONFIRMATION

Table 24: Four-suite held-out VLA panel using the untruncated residual construction: the added residual has nominal rank 72 and the final source-plus-residual adapter has nominal rank 136. Each suite contains two targets and 20 closed-loop episodes per arm. The null-residual arm is the source-chart projection without a chart-external residual.

Suite	Direct	Null residual	Aligned	Random
LIBERO Spatial	10/20	0/20	10/20	0/20
LIBERO Object	6/20	0/20	5/20	0/20
LIBERO Goal	12/20	0/20	13/20	0/20
LIBERO-10	7/20	0/20	8/20	0/20
Pooled	35/80	0/80	36/80	0/80

Against null-residual and matched random, Aligned has 36 wins, zero losses, and 44 ties on paired rollouts. Against Direct, it retains 32 of the 35 successful initializations, or 91.4%, with four Aligned-only and three Direct-only successes. The untruncated construction should equal Direct algebraically. These seven discordances therefore require a loaded-model and inference-consistency audit; nominal rank does not explain them. Neither the one-success pooled difference nor paired retention establishes superiority of an algebraically equivalent update.

Table 25: Retained raw-factor reconstruction audits for the untruncated four-suite construction. Errors compare concatenated BA updates with the candidate's BA, summed across layers. All eight builds record added rank 72 and final nominal rank 136. These are construction-time errors, not effective loaded-weight or action differences.

Suite	Target	Absolute error	Relative error
Spatial	31	7.626923×10^{-6}	2.132361×10^{-7}
Spatial	33	6.890987×10^{-6}	2.129367×10^{-7}
Object	21	7.433086×10^{-6}	2.156451×10^{-7}
Object	22	7.615903×10^{-6}	2.150178×10^{-7}
Goal	12	7.895334×10^{-6}	2.097707×10^{-7}
Goal	16	8.491923×10^{-6}	2.080315×10^{-7}
LIBERO-10	3	7.973017×10^{-6}	2.091478×10^{-7}
LIBERO-10	9	7.075490×10^{-6}	2.079324×10^{-7}

The retained summaries provide these audits and episode-level success arrays. The manuscript package includes their relevant values and source identities in

`figures/source_data/appendix_consistency.json`. Effective adapter scaling, non-adapter weights, same-input actions and repeat-rollout agreement were not independently checked in this revision. The observed differences are not attributed to numerical error without those checks.

Table 26: Source-qualified fresh-target confirmation after rebuilding the source chart. Four selected targets contribute 40 paired closed-loop episodes per arm.

Arm	Direct	Aligned rank 8	Matched random
Pooled successes / episodes	12/40	15/40	0/40

Targets in this panel are selected by source qualification rather than random sampling. We therefore report it separately from Random-20.

K.10 Residual-rank sweep

Table 27: Pooled success in the truncated residual-rank sweep on the same eight VLA targets, with 80 closed-loop episodes per arm. The added rank-8 residual yields a final nominal adapter rank of 72, rather than the untruncated four-suite construction's final rank of 136. The direct column is the fixed rank-8 candidate reference.

Added rank	Aligned	Random	Direct reference	Aligned/Direct
1	4/80	0/80	35/80	11.4%
2	9/80	0/80	35/80	25.7%
4	26/80	0/80	35/80	74.3%
8	35/80	0/80	35/80	100.0%

Both panels use the same eight target IDs and seed-0 rollout initializations. The four-suite panel keeps the untruncated residual (nominal added rank 72), while the sweep truncates it to at most rank 8. Their recorded outcomes are 36/80 and 35/80, respectively, with Direct at 35/80 in both panels. The constructions differ, but their rollout difference is not assigned a cause. Equal seed labels do not establish identical actions or inference randomness. The untruncated update is a candidate-reconstruction reference, whereas the sweep tests lossy residual compression.

Table 28: Direct-success retention in the same rank sweep. This is a paired initialization readout and is not the same quantity as the pooled success ratio in Table 27.

Added rank	Direct successes retained	Direct successes	Retention
1	4	35	11.4%
2	9	35	25.7%
4	16	35	45.7%
8	28	35	80.0%

Rank 8 is the smallest completed rank that passes the predeclared gate of nonzero aligned performance on all eight targets, strict improvement over the matched random arm on all eight targets, and at least 80% of the direct pooled success count. The rank-4 pooled ratio of 74.3% must not be substituted for its paired retention value of 45.7%.

K.11 VLA: stability, continual reuse, and practical cost

The adapter library uses task identity to choose an adapter and stores 8,978,688 bytes after six tasks. Its storage grows with the sequence, unlike the fixed-storage methods. Neither fixed-storage order establishes reliable final retention for aligned refresh.

One-sided paired McNemar values are 0.333 and 0.284 for the two seeds, and 0.208 for their combination. Both seeds favor stabilization, but the differences do not establish statistical superiority.

Table 29: Fixed-storage six-task VLA continuation. Each fixed-storage method stores one rank-72 adapter of 13,379,120 bytes throughout the sequence. Final mean is average success over seen tasks at the last checkpoint; forgetting uses the retained-checkpoint summary's task-wise definition.

Order	Method	Seen-task AUC	Final mean	Mean forgetting
Primary	Aligned	0.1708	0.0167	0.4667
Primary	Direct	0.0958	0.0333	0.2333
Reverse	Aligned	0.1425	0.0000	0.3667
Reverse	Direct	0.1864	0.0000	0.4000

Table 30: Rank-preserving stabilization on two frozen VLA rollout seeds. Each seed contains 39 tasks and 390 episodes per arm. The combined result is directionally positive but not a statistical superiority claim.

Seed	Direct	Stabilized	Direct rate	Stabilized rate	Delta (pp)
20260917	110/390	114/390	28.2%	29.2%	+1.03
20260918	106/390	111/390	27.2%	28.5%	+1.28
Combined	216/780	225/780	27.7%	28.8%	+1.15

Table 31: Common-candidate continuation outcome on 12 method-disjoint LIBERO-90 targets. All arms share the same 5,000-step rank-8 target candidate and then receive the same 50-step continuation budget. This is not an end-to-end comparison from an untouched pretrained model.

Continuation arm	Successes	Rate
Full FT after common candidate	23/120	19.2%
Direct LoRA continuation	22/120	18.3%
Aligned continuation	27/120	22.5%

Table 32: Parameter and resource comparison for the common-candidate continuation. "Requires-grad" counts optimizer-visible parameters; "effective coords" counts degrees of freedom that can change the forward computation. Time is median per target; construction is the target-specific aligned-coordinate construction and serialization. "n/a" means that an arm has no target-specific construction step.

Arm	Req.-grad (M)	Effective (M)	Peak (GB)	Train (s)	Build (s)	Joined (s)
Full FT	392.904	392.904	8.96	26.10	n/a	26.10
Direct LoRA	0.371	0.371	2.47	17.71	n/a	17.71
Aligned continuation	3.342	0.371	2.50	21.74	0.76	22.60

"Aligned continuation" is the arm previously named "Pure Adaptive FTB" in the experiment records: it freezes the retained representation and updates the newly allocated coordinates after the common candidate stage. Relative to Full FT, this arm uses 117.6 times fewer conventional requires-grad parameters and 1,058.1 times fewer effective updated coordinates. Peak logged memory is 72.1% lower. The joined continuation time is 13.4% lower, while training-only time is 16.7% lower. These ratios apply to the continuation after the shared candidate; the 5,000-step candidate stage is common to all three arms and is excluded.

K.12 AUDIT BOUNDARY

Direct candidates provide the absolute-performance reference when complete task-specific adapters can be stored. The mechanism tests compare witness behavior with matched refitting, and aligned directions with matched random capacity. Dynamic chart refresh provides secondary evidence, reported separately from these comparisons.