SUBSPACE-RESTRICTED PLANNING UNDER ACTION COMPRESSION IN LATENT WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Latent world models plan in a learned space, and many of them compress the action before it reaches the predictor. We show that this compression leaves the planner partly blind. On a released Cube-manipulation checkpoint, 15 of 25 action dimensions never reach the predictor, so the planning cost is exactly constant along them. These directions still move the robot. A perturbation along them changes the environment while the model's imagined rollout stays at numerical noise, so no check built on the model can detect it. The planner itself produces this motion. Along a blind direction no candidate looks better than another, so a sampling-based planner executes random motion that it never selected. Raising the sampling budget $27\times$ does not remove this motion. We propose pruned sampling, which restricts the sampler to the directions the cost depends on. One SVD of the frozen weights gives those directions. The restriction needs no training and provably cannot worsen the best achievable plan. On Cube it raises success by 6 points at the standard budget. At that budget, projecting only the executed action gives a gain not distinguishable from this one. At the largest budget tested, pruned sampling keeps its gain and that projection does not. The same restriction extends beyond the exactly blind directions. The spectrum of the second moment of the cost gradient measures how many directions the cost varies along. On four of the five checkpoints studied, this spectrum collapses far inside the visible subspace. Sampling above its spectral gap raises success by 10.7 points on Cube and by 12.3 points on Cube-double. On Cube-double, pruned sampling alone gives no measurable gain. The gains hold for four samplers on Cube and three on Cube-double. A preregistered dose-response test and the same spectrum separate the two effects behind these gains, executed noise and sampling dilution. They tell apart the checkpoints where a restriction helps from the ones where it does not. Project page: https://subspace-restricted-planning.github.io/.

1 INTRODUCTION

A latent world model plans by imagination. An encoder embeds observations into a latent state, and a predictor advances that state under candidate actions. A sampling planner then executes the sequence whose imagined rollout lands closest to the goal Zhou et al. (2024); Maes et al. (2026). The planner sees an action only through the model's action pathway, which in the joint-embedding family compresses each block of consecutive actions through a linear bottleneck. That compression is uneven across directions of the block. When the bottleneck is narrower than the block is wide, some directions reach zero. This null space is the same at every state, whatever the training data. On a released Cube-manipulation checkpoint, 15 of 25 action dimensions never reach the predictor.

Three measured facts describe what this blind subspace does to planning. First, blind perturbations are not physically inert. Their effect on the manipulated object compounds with horizon in all eight Cube probe states (Section 3.2). Second, no check built on the model reports this error. The imagined divergence sits at the float32 floor while the real one does not (Section 3.2). Third, the planner is the main source of these perturbations. Elite selection in a cross-entropy-method (CEM) planner Rubinstein (1999) exerts no pressure on blind coordinates. The executed plan therefore carries an unselected random walk. Its median norm is ≈ 2.9 per block on Cube, which matches a selection-free

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.

analysis (Section 5.2). We call the stock sampler *ambient*. It searches the full action space and spends samples on noise that it then executes.

The same decomposition gives the remedy. One singular value decomposition (SVD) of the frozen action encoder splits the action space into visible and blind subspaces. Restricting the search to visible coordinates provably cannot worsen the best achievable plan (Proposition 4.2). Under block-decomposable proposals the law of the visible search is preserved (Theorem 4.3). Pruned sampling applies this restriction inside the sampler. The execution projection leaves the sampler unchanged and projects only the executed action onto the visible subspace (Section 5.2). On Cube the gains are 6.0 and 6.7 points at the standard budget. Over 300 paired episodes they are not distinguishable. The blind subspace is the exactly-zero block of a broader spectrum. Truncating the search at that spectrum's gap raises Cube by 10.7 points and Cube-double by 12.3.

Our contributions are:

- **The mechanism of blind planning** (Sections 3, 5.2, 5.1). A planner on an action-compressed model executes random motion it never chose, and no check built on the model can see it. We quantify this motion, predict its size from the solver alone, and show that on Cube its removal accounts for the gain of pruned sampling at the standard budget. At a fixed elite count extra candidates provably change nothing along the blind directions (Corollary 4.4). Under the stock protocol, which co-scales the elite count, the gap to the ambient sampler still does not close over a $27\times$ range of budgets.
- **The effective dimension of the planning cost** (Sections 4.3, 5.3). The second moment of the cost gradient measures how many directions the cost varies along. Its spectrum collapses far inside the visible subspace, with a gap at four directions on Cube, three on Cube-double, and two on PushT and Two-Room, whose commands have two dimensions. The architectural SVD yields per-direction drift bounds tighter than scalar baselines at matched coverage (Appendix F).
- **Sampling in the active subspace** (Sections 5.1, 5.3). Truncating the search at the spectral gap raises Cube by 10.7 points and raises Cube-double by 12.3. On Cube-double, pruned sampling alone gives no measurable gain. Cutting through the gap lowers success on both checkpoints. Preregistered dose-response and spectral instruments separate the two channels a subspace restriction acts on, executed noise and sampling dilution (budget spread over cost-flat directions).

2 Related work

Sampling-based planning in latent spaces. Planning by search in a learned latent space goes back to PlaNet Hafner et al. (2019), Dreamer Hafner et al. (2020), and TD-MPC2 Hansen et al. (2024). We study the sampling branch. Within it, iCEM adds colored noise and memory Pinneri et al. (2020), low-discrepancy variants replace random draws Walker et al. (2026), and CEM-GD refines elites by gradients Huang et al. (2021). All change how candidates are drawn. Pruned sampling changes which coordinates vary, and the axes compose (Appendix G).

Low-dimensional action spaces for world-model planning. Several lines shrink what a planner searches: a hand-defined 2D waypoint interface Wang et al. (2026a), an action space learned from video Garrido et al. (2026), and, on the observation side, compact tokenizers Kim et al. (2026). All obtain the space by training or design; we read it off a frozen checkpoint's SVD with an exact invariance statement. Projecting commands into a null space is classical in redundancy resolution, where the null space comes from manipulator kinematics Liégeois (1977); here it comes from a learned model's weights, and a planner free to use it executes noise. PRISM Wang et al. (2026b) reweights where a sampler looks; we delete coordinates the cost cannot see, and the two compose (Appendix G).

Insensitive directions in learned world models. A control-theoretic analysis of latent world models certifies insensitive directions as *safe to ignore* for prediction You et al. (2026). We make the complementary move of deleting them from the *search* and proving the planner loses nothing.

Action insensitivity and active subspaces. Two concurrent works document world models whose predictions ignore actions. ActSWM traces the collapse to the training objective and retrains with dedicated losses Gan et al. (2026); Dueling World Models traces it to a routing degeneracy, repairs the readout on frozen checkpoints, and reports a preregistered null when the repair is converted into planning gains Li et al. (2026). The failure here is a third cause, a capacity zero imposed by the bottleneck, which neither a training incentive nor a readout can recover, so we treat it at planning time. The active-subspace construction behind our task tier comes from uncertainty quantification Constantine et al. (2014); ASEBO applies a related gradient-subspace heuristic in evolution strategies without exactness or closed-loop analysis Choromanski et al. (2019). We bring the construction to world-model planning, where the blind subspace is its exactly-zero case.

Collapse in joint-embedding models. Dimensional collapse of *observation* embeddings is a named failure mode with dedicated diagnostics Jing et al. (2022), and JEPA training guards the observation side Terver et al. (2026). The action pathway carries no such guard. Here it collapses exactly by construction. Large-scale studies of JEPA planning Terver et al. (2026) supply our global-Lipschitz drift-bound baseline. This paper measures what the action pathway's collapse does in closed-loop planning.

3 THE BLIND SUBSPACE

3.1 SETUP AND NOTATION

We study planning with a frozen latent world model of the joint-embedding family Zhou et al. (2024); Maes et al. (2026). An encoder maps an observation window to a latent state; a predictor advances it under an action block $\mathbf{a}_t \in \mathbb{R}^d$, which stacks f control steps of an m-dimensional command, $d = fm$. Actions enter the model through an action encoder whose first layer is a linear map $\mathbf{W} \in \mathbb{R}^{d' \times d}$ applied to the block, and everything downstream of $\mathbf{W}$ is a fixed function of $\mathbf{Wa}$. We write $\mathrm{row}(\mathbf{W})$ for the row space of $\mathbf{W}$ and $\mathcal{N}(\mathbf{W})$ for its null space. The singular values of $\mathbf{W}$ are $\sigma_1 \geq \cdots \geq \sigma_{d'}$ with right singular vectors $v_1, \ldots, v_d$.

A receding-horizon planner selects action sequences by the cross-entropy method (CEM) Rubinstein (1999): sample N sequences $\mathbf{a}_{1:H}$ from a diagonal Gaussian, score them with the model's goal-distance cost $J(\mathbf{a}_{1:H})$, refit to the K lowest-cost candidates, execute the mean after T iterations. The cost depends on actions only through $\mathbf{Wa}_h$.

Trained models compress the action block unevenly across directions. On Cube part of the block is compressed away entirely: $\mathbf{W}$ has rank 10, so the 15-dimensional $\mathcal{N}(\mathbf{W})$ never reaches the predictor, at every state and for any trained weights. This follows from the configuration alone, since any $d = fm > d'$ forces a null space of dimension at least $d - d'$. Action chunking is standard practice in learned control Zhao et al. (2023), so the same argument covers any checkpoint that passes a block through a first *linear* layer narrower than the block.

Of the four released checkpoints of this family, all with $d' = 10$, only Cube is stacked and exactly rank-deficient. The other three are full-rank with compressed spectral tails (Table 2, appendix). The two checkpoints we train with the released recipe, Scene and Cube-double, share the stacked configuration and the same exact deficiency (Section 5.3). The blind subspace is not axis-aligned, so reading it takes the SVD rather than coordinate inspection (Appendix C).

3.2 GROWTH IN THE ENVIRONMENT, INVISIBLE TO THE MODEL

The blind subspace would be harmless if its effect on the environment stayed negligible over a planning horizon. On Cube it does not (Figure 1). We inject matched-magnitude perturbations from 8 probe start states, sustained over a 12-block horizon, along blind, visible, and random directions (instrument and self-checks in Appendix B). Blind-direction deviation grows in 8 of 8 states with no sign of saturation (Figure 1a). The log-slope is $+0.27$ per block (CI $[+0.09, +0.48]$). The $H{=}12$ to $H{=}3$ ratio is 2.2 $[1.4, 4.6]$. Blind growth is slower than visible growth (slope ratio 0.66 $[0.51, 0.74]$), but it is growth, and the deployment metric agrees (Appendix C). On Cube the blind directions move the environment. Section 5.3 tests other checkpoints.

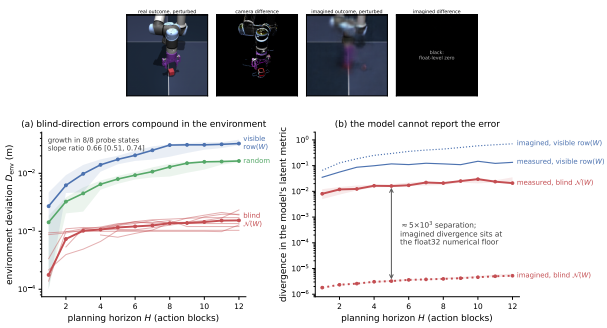


Figure 1: A blind direction moves the environment, not the model. Top: Cube after 12 blocks of a sustained perturbation in $\mathcal{N}(\mathbf{W})$: the camera difference is bright, the imagined one unchanged to float precision (difference panels at $4\times$ gain, Appendix D). **Bottom:** matched perturbations along blind, visible and random directions, medians over 8 start states, interquartile bands, thin lines individual states. **(a)** in the environment, **(b)** through the model.

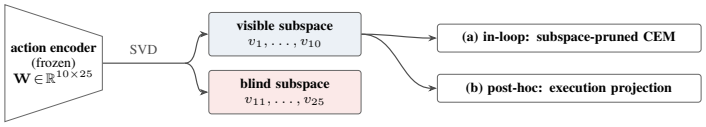


Figure 2: One SVD of the frozen action encoder, two uses. The cost is provably constant along the blind subspace (Proposition 4.2), so restricting the sampler to the visible one preserves the optimum and the law of the visible search (Theorem 4.3). Projecting the executed action reaches the same subspace at actuation time instead.

This compounding error is structurally invisible to any check that consults the model's imagination (Figure 1b). For the same blind perturbations, the divergence measured by encoding the resulting camera frames reaches a median 0.043 in the model's latent metric at $H{=}12$. Rolling those perturbed actions through the predictor instead gives an imagined divergence of 5.2×10^{-6}, at the float32 accumulation floor of 9.3×10^{-6} (Appendix B). The separation is $\approx 8 \times 10^3$ at the full horizon and its median across horizons exceeds 10^3. The architecture guarantees the floor (Proposition 4.2). Our measurement pins its magnitude at deployment precision. The model cannot flag or replan its way out of an error it cannot represent. Any remedy must act on the action side (Section 4.2) or consult environment-side data (Appendix F). A post-hoc decoder renders the dissociation to pixels (Figure 5, appendix).

4 Method

A single SVD of the action bottleneck $\mathbf{W}$ splits the action block space into a subspace the cost can see and one along which it is provably constant (Figure 2). Restricting the sampler to the visible subspace is the method; nothing is trained and nothing in the model is modified.

4.1 Exact cost invariance along the blind subspace

Assumption 4.1. *(i)* The planning cost factors through the bottleneck, that is $J(\mathbf{a}_{1:H}) = \Psi(\mathbf{W}\mathbf{a}_1, \ldots, \mathbf{W}\mathbf{a}_H)$ for some $\Psi : (\mathbb{R}^{d'})^H \to \mathbb{R}$. *(ii)* The proposal distribution has covariance block-diagonal with respect to $\mathrm{row}(\mathbf{W}) \oplus \mathcal{N}(\mathbf{W})$. *(iii)* Elites are the K lowest-cost candidates under a fixed deterministic tie-breaking rule.

Part *(i)* is an architectural fact of these models (Section 3.1). Section 3.2 measures how far it holds at deployment precision. Part *(ii)* holds exactly for the isotropic initialization: throughout for pruned sampling, at the first iteration for the ambient algorithm.

Proposition 4.2 (Cost invariance)**.** *Under Assumption 4.1*(i)*, for any perturbations* $\delta_h \in \mathcal{N}(\mathbf{W})$, $h = 1, \ldots, H$, $J(\mathbf{a}_{1:H} + \delta_{1:H}) = J(\mathbf{a}_{1:H})$.

Proof. $\mathbf{W}(\mathbf{a}_h + \delta_h) = \mathbf{W}\mathbf{a}_h$ for every h, and J is a fixed function of these products.

4.2 Pruned sampling and its guarantees

Let $\mathbf{B} \in \mathbb{R}^{k\times d}$ have orthonormal rows spanning the chosen subspace (Section 4.3 gives the rule). Subspace-pruned CEM restricts the unmodified CEM loop to the coordinates of $\mathbf{B}$: it samples $z \in \mathbb{R}^k$ per block, lifts candidates as $z\mathbf{B}$ for cost evaluation, and refits elite mean and variance in coordinates. An ambient diagonal refit would leak the proposal out of the subspace. With $\mathbf{B}$ the identity the implementation reproduces the stock solver bit for bit under a shared seed, checked by a unit test, so every comparison isolates the restriction.

Theorem 4.3 (Elite decoupling and pruning equivalence)**.** *Consider one CEM iteration under Assumption 4.1. Then* (a) *the elite index set is a measurable function of the candidates' visible components alone;* (b) *the blind components of the K elites are K independent draws from the proposal's blind marginal, carrying no selection pressure; and* (c) *subspace-pruned CEM with $\mathbf{B}$ spanning $\mathrm{row}(\mathbf{W})$ induces the same law on the visible components of that iteration as the ambient algorithm. If* (ii) *holds at every iteration,* (c) *extends by induction to the whole run.*

The proof is elementary (Appendix A): invariance makes the cost a function of visible components only and block-decomposability makes the two components independent, so selection acts on one factor of a product measure.

Corollary 4.4 (Sampling budget cannot buy visibility)**.** *Under the conditions of Theorem 4.3, the law of the elites' blind components does not depend on the number of samples N.*

The invariance is planner-agnostic; Appendix A derives what MPPI and a gradient planner inherit.

Executed blind noise. Because elites carry no selection pressure in $\mathcal{N}(\mathbf{W})$, the blind component of the CEM mean performs an unbiased random walk. Its scale contracts only through resampling. We simulate this recursion as the solver does. In the standard configuration (K=30, T=30, 15 blind dimensions) it predicts an executed blind norm of $\approx$2.95 and a contraction of $\approx$0.46, both elite-count dependent (chain and budget grid in Appendix A). Section 5.2 tests both predictions.

Scope of the theorem. The theorem says pruning gives up nothing, not that the two planners perform equally: the ambient planner dilutes samples over d dimensions rather than k and executes the blind walk above.

4.3 Choosing the subspace: two tiers

The blind subspace of $\mathbf{W}$ is the exactly-zero case of a more general object. Let $\mathbf{C} = \mathbb{E}_\rho[\nabla_{\mathbf{a}} J\, \nabla_{\mathbf{a}} J^\top]$ be the second moment of the cost gradient over action blocks, taken under the planner's proposal distribution ρ, with eigenvalues $\lambda_1 \geq \cdots \geq \lambda_d$. This is the active-subspace construction of Constantine et al. (2014). A small trailing eigenvalue means the cost barely varies along that direction. Because J depends on the block only through $\mathbf{W}\mathbf{a}$, the chain rule gives $\mathbf{C} = \mathbf{W}^\top \mathbf{M} \mathbf{W}$ for a positive-semidefinite $\mathbf{M}$, so $\mathcal{N}(\mathbf{W}) \subseteq \mathrm{null}(\mathbf{C})$ and $\mathrm{rank}\,\mathbf{C} \leq d'$. Proposition 4.2 recovers the case where the trailing eigenvalues vanish identically. When they are small rather than zero, the active-subspace bound of Constantine et al. (2014) applies. The cost deviates from its conditional average over the top-k eigendirections by at most $\sum_{i>k} \lambda_i$ in mean square, with constant 1 for Gaussian ρ. The CEM proposal is Gaussian, so the constant holds under the sampler's own measure. The bound controls cost approximation, not closed-loop success. Section 5.3 supplies that link empirically.

The rule for choosing $\mathbf{B}$ therefore has two tiers. The architectural tier prunes $\mathcal{N}(\mathbf{W})$, which Proposition 4.2 makes lossless at every state. The task tier truncates at a spectral gap of $\mathbf{C}$. Estimating $\mathbf{C}$ takes a few hundred backpropagated gradients of the frozen cost, and the estimate's error shrinks with the gap Constantine et al. (2014). A gapless spectrum gives no principled cut, and the task tier does not apply there (Section 5.3 shows one such checkpoint). On the two full-rank checkpoints we evaluate we instead prune $\mathbf{W}$'s weakest singular directions (σ-tail heuristic), a count set per environment. The paired ablation in Section 5.1 finds no cost above noise on either checkpoint. Appendix E deletes the same directions from the PushT model instead of from the sampler (spectral surgery). The measure ρ shifts as CEM refits. Section 5.3 checks empirically that the estimated subspace is stable under this shift.

Table 1: Closed-loop success across environments and budgets. Success rates (%, mean±sd over 3–6 seeds, 50 episodes per seed) and paired differences, with the exact McNemar p-value in parentheses. The active row samples the top eigendirections of $\mathbf{C}$ (k=4 on Cube, k=2 on PushT, chosen at the spectral gap before any closed-loop run). Controls run on Cube, where the blind subspace is exact.

	Cube (k=10 of d=25)			PushT (k=7 of d=10)		
sampler \ N	30	100	300	30	100	300
ambient CEM	56.7 ±8.1	64.0 ±8.0	63.0 ±7.1	76.0 ±5.3	85.3 ±3.1	88.0 ±4.0
pruned (ours)	56.7 ±3.1	67.3 ±9.9	69.0 ±6.5	79.3 ±5.0	87.3 ±4.2	89.3 ±1.2
active (ours, spectral k)	—	—	73.7 ±4.6	—	—	86.0 ±2.0
random subspace	50.0 ±2.0	52.7 ±6.1	58.7 ±9.2	—	—	—
blind subspace	45.3 ±5.0	38.0 ±3.5	37.3 ±7.0	—	—	—
Δ pruned−ambient	+0.0 (1.000)	+3.3 (.302)	+6.0 (.005)	+3.3 (.511)	+2.0 (.629)	+1.3 (.824)
Δ active−ambient	—	—	+10.7 (¡.001)	—	—	−2.0 (.664)

The SVD of $\mathbf{W}$ also yields bounds on drift, the distance between the environment and the imagined rollout after H blocks. We fit the drift per horizon from the perturbation's energy in each singular band of $\mathbf{W}$ and wrap the fit in a split-conformal layer Vovk et al. (2005). With eight to sixteen probe states the coverage is empirical rather than certified (Appendix F).

5 Experiments

Protocol. We evaluate on five benchmarks. Cube Park et al. (2024), PushT Chi et al. (2024) and Two-Room use the released frozen checkpoints of Maes et al. (2026). Scene and Cube-double (two cubes, joint success) have none, so we train each with the released recipe (100 epochs, 3,000 collected episodes against the released 10,000). Scene's open-loop prediction error is below the released Cube model's, and Cube-double's protocol carries a measured replay ceiling of $86.7 \pm 1.2\%$ (Appendix B, spectra in Table 2). Cube, Scene and Cube-double are pruned to their row spaces, PushT to k=7 and Two-Room to k=9 by the σ-tail heuristic of Section 4.3. Each run evaluates 50 episodes from the benchmark dataset. Comparisons pair episodes by start state and seed and use exact McNemar tests (b_{01}/b_{10} = pairs improved/degraded by the first-named sampler), and gains must be seed-consistent. An action block stacks f=5 control steps of an m-dimensional command (m=5 on the manipulation checkpoints, m=2 elsewhere), standardized per dimension; one block executes between replans. CEM uses T=30 iterations and $K=N/10$ elites; the standard budget is N=300 and the sweep co-scales both. Seed counts are in Appendix I. All statistics are recomputed from stored per-episode records.

5.1 Both tiers of the restriction improve Cube at standard budgets

Both tiers are tested against the ambient sampler at the standard budget N=300 (Table 1). The architectural tier, sampling in $\mathrm{row}(\mathbf{W})$, raises Cube from 63.0% to 69.0%. That is +6.0 points over 300 paired episodes (b_{01}=28, b_{10}=10, $p = 0.0051$). The gain survives Bonferroni, with five of six seeds improved and none degraded. The task tier samples the top four eigendirections of $\mathbf{C}$ (the active rows of Table 1), 4 of 25 dimensions. It raises Cube to 73.7%, a gain of +10.7 points (b_{01}=39, b_{10}=7, $p < 0.001$, all six seeds positive). Against the architectural tier directly, the task tier gains +4.7 in the pooled pairing ($p = 0.02$) with per-seed differences mixed in sign, so we claim only that it is no worse. The architectural gain shrinks at N=100 (+3.3, $p = 0.30$) and vanishes at N=30, where flips are exactly symmetric. The gap emerges with budget and persists (mechanism in Section 5.2).

Two controls attribute the gain to this subspace. A random 10-dimensional subspace is *worse* than ambient (−11.3 at N=100, $p < 0.001$), so the gain is not a dimensionality effect. The blind subspace collapses success to 37–45% ($p \leq 0.002$ at every N), so the visible subspace carries the control.

The two full-rank checkpoints, whose pruned fractions are far smaller, move less. PushT shows no loss at any budget and no significant difference (Table 1). Two-Room, pruned at its weakest direction, goes from 87.3% to 88.7% over 150 pairs and three seeds (+1.3, b_{01}=6, b_{10}=4, $p = 0.75$, seeds mixed in sign). Their task tier samples 2 of 10 dimensions. It moves PushT by −2.0 points, CI

$[-8.0, +4.0]$ (Table 1 gives the exact p), and leaves Two-Room's flips exactly symmetric, seven each way. Spectral surgery deletes PushT's three pruned directions from the model instead of from the sampler, and aggregate success is unchanged over 150 paired episodes (Appendix E). Among these checkpoints the gain is specific to Cube, and Section 5.3 finds that exact blindness does not predict it.

Sampling budget does not buy the difference back. Corollary 4.4 holds at fixed elite count; the stock protocol instead co-scales $K = N/10$, so raising the budget also thins the executed blind component. The exact chain of Appendix A predicts its norm falling from 2.95 at N=300 to 1.23 at N=2700, near the amplitude where injected blind noise produced zero flips. The sweep therefore favors the ambient sampler, which gets both more candidates and less blind noise. Over a $27\times$ range (Figure 8c) both samplers improve slowly, as visible-marginal optimization predicts. At N=2700 the ambient sampler reaches 69.3% over six seeds against 73.3% for pruned sampling, and the paired gap does not close. It is $+4.0$ points at the matched budget (b_{01}=15 against b_{10}=3, $p = 0.0075$, all six seeds positive), so the predicted $2.4\times$ thinning does not buy it back. An elite-count control separating K from N points the same way: the gap narrows as larger K thins the predicted noise but stays positive at every K tested (Appendix A).

5.2 The planner's internals match the analysis

Instrumenting the solver matches both quantitative predictions of Section 4.2 (Figure 6). Blind coordinates contract by $\times 0.41$ over 30 iterations, at the selection-free chain's $\times 0.46$ and far from pruned sampling's $\times 0.19$. Visible coordinates contract by a nearly identical $\times 0.37$. Elite evidence diluted over 25 dimensions sharpens selection only weakly (the dilution channel, isolated next). The executed blind component has median norm 2.89 per block (P90 3.90, n=250) against the predicted 2.95: within 2% in magnitude and 12% in contraction. An independent pipeline reading the executed commands themselves reproduces the median at 2.87 (Appendix C).

A dose grid tests that level against injected noise. Amplitude 1.0 flips 0 of 120 paired episodes. Amplitude 4.0 drops mean success 7.5 points (grid in Appendix C).

A decomposition experiment settles which channel carries the gain. We run the ambient sampler unchanged and only project the *executed* action onto $\mathrm{row}(\mathbf{W})$ at actuation time, leaving every sample, cost, and elite refit untouched. We call this arm the execution projection. It is the mechanism-isolating control, and Theorem 4.3 explains why it loses nothing. It recovers the full gain of pruned sampling, $+6.7$ points over the ambient sampler (300 paired episodes, six seeds, b_{01}=24, b_{10}=4, $p = 0.0002$, every seed positive). Pruned sampling adds nothing measurable beyond it (-0.7, $p = 0.86$, seeds mixed). At the standard budget the Cube gain therefore comes from the executed-noise channel. The deeper visible contraction of pruned sampling (Figure 6a) does not convert into success. Quadrature reconciles the dose grid with the gain. Adding amplitude 1.0 atop the planner's own 2.9 moves the total only to 3.07, whereas removing the entire 2.9 is a large change.

The decomposition is budget-dependent as the noise account predicts. At N=2700 the chain puts the executed blind norm at 1.23, near the zero-flip range. Execution projection recovers $+1.7$ ($p = 0.27$) while pruned sampling holds $+4.0$ ($p = 0.0075$). The noise-removal gain shrinks with the predicted noise. The residual gap of pruning over projection, $+2.3$ with all six seeds positive but $p = 0.12$, is consistent with the in-loop restriction mattering once candidates are plentiful; we leave its attribution open.

5.3 Two channels decide where the restriction pays

Scene and Cube-double share Cube's exact blind spot and replicate the mechanism: executed blind walks of median norm 2.83–2.98 per block against the same selection-free prediction of 2.95, and a null-space sampler that collapses success in each (Table 6). The mechanism transfers; the payoff does not.

On Scene the pruned$-$ambient difference is -3.7 points against a 77.3 ± 4.3 baseline (b_{01}=15, b_{10}=26, CI $[-8.0, +0.3]$, p=0.117), and the execution projection moves -1.0 (p=0.75). The blind subspace still carries real control. The null-space sampler collapses to 56.0%, and a random subspace of the same rank never beats ambient in any seed. The environment explains the null result. In four of five probe states, matched blind, weak-visible, and random perturbations leave the scored objects

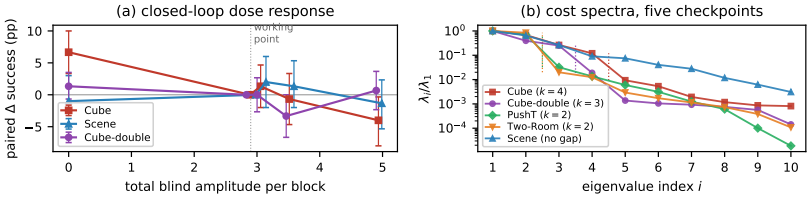


Figure 3: Two instruments separate the noise channel from the search channel. **(a)** Dose response: paired success change against total blind amplitude per block, from full removal (leftmost point) through each planner's working point (dotted) to $1.7\times$ beyond; 95% bootstrap bars. **(b)** spectra of $\mathbf{C}$, the cost-gradient second moment, with gaps marked. The spectrum of $\mathbf{W}$ is gradual on every checkpoint (Table 2).

exactly still. In 47% of episodes the gripper never makes contact, and there paired flips are exactly symmetric. On these checkpoints, coupling between blind motion and the scored objects is necessary for a gain.

It is not sufficient. On Cube-double the scored objects respond to blind perturbations in six of eight probe states, at amplitudes about a fifth of the blind motion the planner actually executes. We preregistered a positive prediction and fixed the disposal rule before any closed-loop run. It failed. Pruning changes success by $+0.0$ points over 300 paired episodes (16 flips each way, CI $[-3.7, +3.7]$), and the execution projection pools to $+1.3$ over 600 pairs across twelve seeds (CI $[-0.7, +3.5]$; Appendix C). A gain of Cube's size is excluded; a smaller one is not resolvable at this sample size.

Two probes rule out the obvious framing artifacts. Re-scoring every stored episode at a swept threshold leaves the null result in place at every τ (Figure 7). A nine-fold ambient budget raise moves Cube-double $+2.0$ points against $+6.3$ on Cube.

The two null results have different mechanisms, and two preregistered instruments separate them (Figure 3). The first sweeps the executed-noise channel. Injected blind noise moves each environment along its own planner's amplitude axis. On Cube the response is steep on both sides of the working point, $+6.7$ from removal down to -4.0 at the largest dose (marginal at six seeds, p=0.058, in the direction of the Section 5.2 dose grid). On Scene and Cube-double the curve is flat, both under half of Cube's drop at the largest dose, as preregistered. Removing executed noise pays where this curve is steep.

The second instrument measures what the search itself faces: the spectra of $\mathbf{C}$ split the five checkpoints into three shapes. Cube is gapped at four directions. Cube-double collapses to three ($13\times$ gap, fourth eigenvalue at 1.9% of the first). Scene is gapless. The two full-rank checkpoints collapse to two ($22\times$ and $42\times$ gaps), which equals their command dimension. The directions above each gap can be described without the planning cost. Blocks holding one command across all five control steps span the action-repeat subspace, one direction per command. On all four gapped checkpoints every eigendirection of $\mathbf{C}$ above the gap has at least 0.937 of its squared norm there (Appendix H). Run on out-of-family TD-MPC2 locomotion checkpoints the same instrument finds no gap on any of three tasks and rules truncation out.

Sampling the top three eigendirections converts the Cube-double diagnosis into success. It gains $+12.3$ points over ambient (b_{01}=49, b_{10}=12, $p<0.001$, all six seeds positive) and $+12.3$ over $\mathrm{row}(\mathbf{W})$, for 45.0% under the $86.7 \pm 1.2\%$ ceiling. On Cube-double, removing the blind subspace paid nothing and removing the visible flat remainder pays the most. Cutting through the gap is punished on both, 26.0 points on Cube (zero episodes improved) and 10.7 on Cube-double (p=0.0025), while Scene's gapless spectrum gives the rule no cut, so the task tier is not applied.

We ran an action-repeat sampler as a preregistered control, on the components the spectrum keeps (four on Cube, three on Cube-double) and on all five. The task tier compares differently with this control on the two checkpoints. On Cube, neither version of the control is distinguishable from the task tier at 300 pairs (Table 8). On Cube-double the action-repeat sampler is the stronger of the two, by 3.3 points over 300 pairs. The components the block keeps decide the gap, and holding all five

loses 10.0 points against holding three (Table 8). Where the two are not distinguishable, we cannot resolve at this sample size what the spectrum adds beyond the block layout and the drop list.

Retraining Cube from scratch keeps the gain but not the gap. Across three retrained checkpoints the top-four restriction stays positive in nine of nine paired seeds. The gap itself shallows and shifts enough on one retrain that the criterion picks a harmful k (Appendix H), so the spectra and k values above are properties of the released checkpoints. A freshly trained three-block checkpoint matches Cube-double's gap position, post-gap mass, and flat-direction count, yet its own gap cut pays nothing (-1.3, n.s.): the spectral profile alone does not decide the payoff (Appendix H).

Removal is lossless by Proposition 4.2 on the three checkpoints with a null space, and costs nothing measurable on the two full-rank ones. The payoff is measured channel by channel: the noise channel pays where the dose response is steep (Cube), the search channel where the spectrum collapses inside the visible subspace (Cube-double), and neither where blind motion never reaches the scored objects (Scene). All instruments ran preregistered with fixed disposal rules, and the restriction transfers across samplers. Against matched ambient baselines, the task tier gains $+20.0$ (MPPI), $+10.7$ (shooting), and $+18.7$ (iCEM) on Cube, and $+16.7$ (MPPI) and $+14.0$ (shooting) on Cube-double. The executed blind noise falls as effective sample count grows, as Appendix A predicts (Appendix H). Three further tests use the same decompositions and are reported in Appendices E, F and G.

6 Limitations and conclusion

The evidence is scoped in eight ways. First, all five benchmarks are simulated and every model carrying the exact blind space comes from one family with a linear first action layer, which is all the invariance argument needs. Effect sizes do not transfer even within the family (Section 5.3), and hardware is untested. No gain under the spectral criterion outside this family is yet observed. On a shallow-gap retrain the criterion selects a harmful k, so gap-based k selection should be trusted only where the gap is deep. A checkpoint matching Cube-double's spectral profile gains nothing from its own gap cut, so no single spectral statistic we measured predicts the dilution payoff (Appendix H). Second, the drift bounds are calibrated on sixteen (Cube) and eight (PushT) probe states. Coverage averages slightly below nominal, and PushT widths vary with the split (Appendix F). Third, the ambient sampler meets Assumption 4.1*(ii)* only at its first iteration (the residual coupling is measured, not bounded), so causal attribution in Section 5.3 rests on the execution projection. Every execution projection here targets $\mathrm{row}(\mathbf{W})$ of the model in use. Projecting onto a strict subspace of it would discard visible components, which elite selection acts on (Theorem 4.3), and no result here measures that cost. Fourth, the σ-tail heuristic is checked per environment by paired ablation. It gives no universal threshold. Fifth, the prior combination is evaluated on one checkpoint, where the prior dominates (Appendix G). Sixth, the task tier's guarantees are partial. The active subspace, estimated under the initial proposal measure, moves by at most 7 degrees under the final elite distribution (Appendix H). No theory covers the shifting measure. The Poincaré bound controls cost approximation, not closed-loop success. We read effective dimension as the task's controllable degrees of freedom, a reading consistent with five checkpoints and not tested by them. Seventh, the action-repeat control gave different outcomes on Cube and Cube-double (Section 5.3); where it matches the task tier at 300 pairs, a difference inside the reported interval remains possible, and the comparison covers CEM at the standard budget. The task-tier advantage over the architectural tier on Cube is pooled-significant but seed-mixed. The trained checkpoints use 3,000 episodes against the released 10,000, Cube-double rates sit under the $86.7 \pm 1.2\%$ replay ceiling, and Scene's mixed criterion leaves its gain-threshold profile undefined. Finally, the exact null space could be removed at training time by widening the bottleneck to $d' \geq d$; training at fixed width can only orient it (Section 3.2). The analysis targets frozen checkpoints and measures what the restriction recovers without that fix.

A frozen world model's action encoder states, in its weights, which action directions its cost can never see. A sampling planner over such a model executes noise it never chose, no imagination-side check can detect it, and no budget shrinks it at fixed elite count. On Cube, removing it is the entire measurable benefit of pruned sampling at the standard budget. Two further checkpoints with the same blind subspace replicate the mechanism without the benefit. The gradient spectrum separates the two, and sampling above its gap turns the larger null result into a 12.3-point gain. For the models studied, one SVD specifies what a planner can drop without loss, one backward pass how few directions the cost varies along, and two preregistered instruments where removal works.

AI USE STATEMENT

We used generative AI tools only to polish and draft manuscript text. We did not use them to design or refine experiments, to implement methods or analysis code, or to run experiments. We did not use them to form hypotheses, to build the theoretical model, to state the mathematical claims, or to draft the proofs. We did not use them for data cleaning, for qualitative analysis, or for the interpretation of results. We built no synthetic data set and translated no text, so those two tasks do not apply. We reviewed and edited all AI-drafted text ourselves. No statistic is taken from the output of a generative AI tool. Every number is recomputed from stored records by the scripts in the supplementary material. We take responsibility for the final content of this work, including text produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Appendix I maps every result in the paper to the instrument that produced it and lists seed counts and runtimes. The supplementary material contains the code, the per-episode records behind every number, and the scripts that recompute each figure, table, and statistic from those records. A CPU-only self-check runs them end to end, and the environment is pinned. The supplementary archive also carries the project page as two self-contained HTML files, with every chart, figure and video clip embedded; they open from the archive with no network. The same page is hosted anonymously at `https://subspace-restricted-planning.github.io/`. The LeWM checkpoints and datasets Maes et al. (2026) are public. The checkpoints and datasets we produced are too large for the supplementary archive. Every one of them is trained or collected with the released recipe on a public benchmark, and the included README names each with its size and its recipe. Links to the hosted copies follow at camera-ready. Appendix A contains the proofs, and Appendix B the full measurement protocol.

REFERENCES

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *arXiv preprint arXiv:2303.04137*, 2024.

Krzysztof Choromanski, Aldo Pacchiano, Jack Parker-Holder, Yunhao Tang, and Vikas Sindhwani. From complexity to simplicity: adaptive ES-active subspaces for blackbox optimization. In *Advances in Neural Information Processing Systems*, 2019.

Paul G. Constantine, Eric Dow, and Qiqi Wang. Active subspace methods in theory and practice: applications to kriging surfaces. *SIAM Journal on Scientific Computing*, 36(4):A1500–A1524, 2014.

Yuhao Gan et al. ActSWM: Action-sensitive world models for long-horizon planning in open-world games. *arXiv preprint arXiv:2607.26712*, 2026.

Quentin Garrido, Tushar Nagarajan, Basile Terver, Nicolas Ballas, Yann LeCun, and Michael Rabbat. Learning latent action world models in the wild. *arXiv preprint arXiv:2601.05230*, 2026.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. *arXiv preprint arXiv:1811.04551*, 2019.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2020.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. *arXiv preprint arXiv:2310.16828*, 2024.

Kevin Huang, Sahin Lale, Ugo Rosolia, Yuanyuan Shi, and Anima Anandkumar. Cem-gd: Cross-entropy method with gradient descent planner for model-based reinforcement learning. *arXiv preprint arXiv:2112.07746*, 2021.

Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2022. arXiv:2110.09348.

Dongwon Kim, Gawon Seo, Jinsung Lee, Minsu Cho, and Suha Kwak. Planning in 8 tokens: A compact discrete tokenizer for latent world model. *arXiv preprint arXiv:2603.05438*, 2026.

Ang Li, Ryan Fei, Weiyu Zhou, and Kenta Hayashi. Dueling world models: advantage-style action channels for common-mode distractor rejection. *arXiv preprint arXiv:2608.06706*, 2026.

Alain Liégeois. Automatic supervisory control of the configuration and behavior of multibody mechanisms. *IEEE Transactions on Systems, Man, and Cybernetics*, 7(12):868–871, 1977.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. LeWorld-Model: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026.

Christian Oefinger, Finn Rasmus Schäfer, Korbinian Moller, Mattia Piccinini, and Johannes Betz. Validate the dream before you trust its verdict: Admissibility for world-model simulators. *arXiv preprint arXiv:2607.07196*, 2026.

Seohong Park, Kevin Frans, Benjamin Eysenbach, and Sergey Levine. OGBench: Benchmarking offline goal-conditioned RL. *arXiv preprint arXiv:2410.20092*, 2024.

Cristina Pinneri, Shambhuraj Sawant, Sebastian Blaes, Jan Achterhold, Joerg Stueckler, Michal Rolinek, and Georg Martius. Sample-efficient cross-entropy method for real-time planning. *arXiv preprint arXiv:2008.06389*, 2020.

Reuven Y. Rubinstein. The cross-entropy method for combinatorial and continuous optimization. *Methodology and Computing in Applied Probability*, 1(2):127–190, 1999.

Jaeuk Shin, Yoonseok Ra, and Insoon Yang. From prediction uncertainty to conformalized distance fields for safe motion planning. *arXiv preprint arXiv:2607.00776*, 2026.

Basile Terver, Tsung-Yen Yang, Jean Ponce, Adrien Bardes, and Yann LeCun. What drives success in physical planning with joint-embedding predictive world models? *arXiv preprint arXiv:2512.24497*, 2026.

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

Markus Walker, Daniel Frisch, and Uwe D. Hanebeck. Sample-efficient and smooth cross-entropy method model predictive control using deterministic samples. *arXiv preprint arXiv:2510.05706*, 2026.

Alex N. Wang, Trevor Darrell, Pavel Izmailov, Yutong Bai, and Amir Bar. Lifting embodied world models for planning and control. *arXiv preprint arXiv:2604.26182*, 2026a.

Hongbo Wang. Certified world models: Predictability across configuration, horizon, and resolution. *arXiv preprint arXiv:2606.13092*, 2026a.

Hongbo Wang. Conformal orbit-valid trust horizons for equivariant world models. *arXiv preprint arXiv:2606.24946*, 2026b.

Yuhai Wang, Jiawei Xia, Rongxuan Zhou, Xiao Hu, Yongliang Shi, Jing Du, and Yang Ye. Prism: Prior-guided imagination sampling in world models. *arXiv preprint arXiv:2606.07974*, 2026b.

Hanzhe You, Yonggang Zhang, Maohao Ran, Zhiqin Yang, Zhenyuan Zhang, Wei Xue, Jun Song, Xinmei Tian, and Yike Guo. A control theory of predictability in latent world models. *arXiv preprint arXiv:2607.10362*, 2026.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems (RSS)*, 2023. arXiv:2304.13705.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2024.

A PROOFS

Setting. Write $\mathbf{\Pi}_{\text{row}}$ and $\mathbf{\Pi}_{\mathcal{N}}$ for the orthogonal projections onto $\text{row}(\mathbf{W})$ and $\mathcal{N}(\mathbf{W})$. Fix one CEM iteration with N candidates $\mathbf{a}^{(i)} = v^{(i)} + n^{(i)}$, $i = 1, \dots, N$, where $v^{(i)} = \mathbf{\Pi}_{\text{row}}\mathbf{a}^{(i)}$ and $n^{(i)} = \mathbf{\Pi}_{\mathcal{N}}\mathbf{a}^{(i)}$, applied per block, the horizon index playing no role. Assumption 4.1*(ii)* makes the proposal block-decomposable, so $(v^{(i)})_i$ and $(n^{(i)})_i$ are independent families. By Assumption 4.1*(i)* there is a function $\tilde{J}$ with $J(\mathbf{a}^{(i)}) = \tilde{J}(v^{(i)})$, and by Assumption 4.1*(iii)* the elite index set $S \subset \{1, \dots, N\}$ is determined by the costs alone.

Proof of Theorem 4.3. (a) S is a deterministic function of $(J(\mathbf{a}^{(1)}), \dots, J(\mathbf{a}^{(N)})) = (\tilde{J}(v^{(1)}), \dots, \tilde{J}(v^{(N)}))$, hence $\sigma\big((v^{(i)})_i\big)$-measurable, and therefore independent of $(n^{(i)})_i$.

(b) Let s be any admissible elite set. Since S is $\sigma\big((v^{(i)})_i\big)$-measurable and $(n^{(i)})_i \perp (v^{(i)})_i$,

$$\mathcal{L}\big((n^{(i)})_{i\in S} \,\big|\, S = s\big) = \mathcal{L}\big((n^{(i)})_{i\in s}\big) = \bigotimes_{i\in s} P_n,$$

where P_n is the proposal's blind marginal. The elites' blind components are thus K independent draws from P_n regardless of the realized S: selection carries no information about them.

(c) Couple the two algorithms by sharing the visible sample stream. At the first iteration both draw visible components from the same marginal (the pruned algorithm draws them directly in coordinates, and the ambient algorithm draws them as the visible block of its product proposal), so the joint law of $(v^{(i)})_i$ agrees. Costs agree pointwise by $J = \tilde{J} \circ \mathbf{\Pi}_{\text{row}}$, so the elite set has the same law, and the refitted visible mean and (diagonal, in-coordinates) variance are the same measurable function of $(v^{(i)})_{i\in S}$ in both algorithms. By induction, as long as the ambient proposal remains block-decomposable, every iteration's visible proposal parameters, candidates, elites, and returned visible mean agree in law. The pruned algorithm never represents blind coordinates, and by *(b)* removing them does not alter any visible-component distribution.

Proof of Corollary 4.4. By part *(b)* the elites' blind components are K draws from the proposal blind marginal P_n, whatever the candidate count N. N enters only through the elite *index* distribution, which is independent of the blind components.

The executed blind component. Under the ambient algorithm, part *(b)* makes the blind refit a two-line recursion. The blind mean is refit to the average of K i.i.d. draws centered at the incumbent mean with scale σ_t, so it performs an unbiased random walk accumulating variance σ_t^2/K per dimension per iteration. Over T iterations the executed blind component has expected squared norm $\dim\mathcal{N}(\mathbf{W}) \cdot \sum_{t=1}^{T} \sigma_t^2/K$. The blind scale itself contracts without selection: the sample standard deviation of K i.i.d. Gaussians satisfies $(K-1)s^2/\sigma^2 \sim \chi^2_{K-1}$, giving the per-iteration log-drift $\mathbb{E}[\ln s - \ln\sigma] = \frac{1}{2}\big[\psi(\frac{K-1}{2}) + \ln 2 - \ln(K-1)\big] \approx -0.0174$ for K=30, deflated further by the solver's retention of the incumbent mean as candidate 0, which pins one elite's blind component at the incumbent whenever the incumbent survives selection. We obtain the prediction by simulating this recursion exactly as the solver implements it (unbiased per-dimension standard-deviation refit, incumbent pinned; `ek3_theory.py` in the supplementary material). The simulation assumes the incumbent is elite at every iteration; dropping that assumption gives an upper envelope. At the experimental configuration (K=30, T=30, unit initial scale, 15 blind dimensions per block) the exact chain predicts executed norm 2.95 per block and total contraction 0.46, the two constants tested in Section 5.2. Both are tested out of sample three times: the executed medians are 2.89, 2.98, and 2.83 per block on Cube, Scene, and Cube-double, with contractions 0.41, 0.44, and 0.48 (Table 6). The closed-form chain built from the deterministic log-drift recursion is biased low by Jensen's inequality (it predicts ≈2.5 and 0.36) and we use it only as an approximation. Both constants depend on the elite count: over the co-scaled budget grid of Section 5.1, K=10/30/90/270 gives predicted executed norms 3.42/2.95/2.00/1.23, and the elite-count control at N=300, K=100 is predicted at 1.93. Measured, that control narrows the paired gap from +6.0 to +3.7 (b_{01}=20, b_{10}=9, $p = 0.061$) while the ambient arm's own change is not resolvable (+1.7, $p = 0.53$); the gap stays positive at K=10, 30, 100, 270 and significant at K=30 and 270. Figure 9 plots the measured executed norms of this elite-count family against the paired gains, beside the temperature and shooting families of Appendix H.

Other planners. The invariance is planner-agnostic. The budget-independence is not. MPPI weights candidates by $\exp(-J/\lambda)$, which blind components cannot change, so its plan also carries unselected blind noise. Each iteration's blind increment is a weighted average over all N candidates, with variance σ^2/n_{eff}; because MPPI never contracts its sampling variance, the increments accumulate as a random walk over the T iterations, whose level falls as n_{eff} grows, the scaling the temperature sweep of Appendix H measures. Corollary 4.4 is specific to elite refitting, whose blind refit sees only the K elites, and no number of candidates changes it at fixed elite count. A gradient planner receives exactly zero true gradient along the blind subspace. Finite differences of the cost along blind directions return identically zero at deployment precision, while autograd returns only a backward-precision residual, four orders below the visible gradient (Appendix B). Blind coordinates of a gradient planner therefore freeze at initialization rather than walk.

B MEASUREMENT PROTOCOL

We list the final protocol with one-line rationales. Per-sample raw values are stored for every quantity.

- *Determinism.* Environments are rebuilt from full state at each replay. Repeating identical inputs changes Cube outcomes by at most 1.8×10^{-15} (4/4 states exact or ULP-level), and PushT replays are exact after state scrubbing. Cross-*process* replays of contact-rich PushT episodes can differ at the single-episode level, so all paired comparisons run within one process and re-collected records are deduplicated by tag.
- *Blindness floor (used throughout as "the float32 floor").* Injecting $\mathcal{N}(\mathbf{W})$ perturbations changes the imagined latent by 2.4×10^{-6} (single-step) and 9.3×10^{-6} (12-block sustained), consistent with float32 accumulation of $\mathbf{W}\delta$ residuals. The visible-direction response at matched magnitude is 8.5×10^{-2}, about four orders larger. An in-run monitor asserts this separation for every collection batch, and holds on the trained checkpoints as well, with separations of 8.4×10^{-4} (Scene) and 4.2×10^{-4} (Cube-double) against the 10^{-3} gate.
- *Interior-point injection.* On PushT, perturbation windows are pre-screened so that injected actions remain strictly inside actuation bounds. Cube actions sit at the bounds in almost every window, leaving no window to screen, so there the reference block is clipped instead to leave each perturbation exactly enough room. Clip counts are recorded per batch and were zero in the growth measurements. Action bounds are taken from the environment. The dataset contains actions outside these bounds.
- *Why comparisons are paired.* Reseeding the planner's own sampling RNG perturbs trajectories about $3\times$ more than injected blind noise, so the injection channel cannot produce large single-episode effects. All comparisons are therefore paired by start state and seed and run over multiple seeds.
- *Surgery seed sets.* Success rates in Appendix E come from the surgery protocol and its seed sets, with the intact model at 87.2% over twelve seeds. The healthy-model numbers of Table 1 come from the independent healthy-protocol collection, and the two protocols agree within seed noise.
- *No selective truncation.* Success-triggered episode termination is disabled during measurement rollouts, since terminating on success would censor exactly the trajectories with large deviations.
- *Pairing and tests.* Comparisons pair episodes by dataset start state and seed. We report exact McNemar tests on paired success flips and require seed-consistency before interpreting a gain.
- *Probe extension for drift bounds.* The Cube calibration set extends the growth collection with eight additional start states, disjoint from the original eight, each probed at three perturbation amplitudes ($\varepsilon \in \{0.25, 0.5, 1.0\}$) along the sustained arm. All collections pass the in-run separation monitor; margin-clip touches at most 52 of 300 action values per state at the largest amplitude, recorded per state.
- *Gradient check.* The directional derivative of the cost along blind directions, measured by finite differences in the differentiable imagination pipeline, returns identically zero at both tested step sizes. The autograd gradient carries a blind component at 8×10^{-4} of the visible gradient in relative terms, identified as backward float32 accumulation rather than signal.

- *Solver equivalence.* With identity basis, the subspace solver reproduces the stock solver bit for bit under a shared seed, asserted as a unit test. Injected closed-loop perturbations are verified at the actuator interface to 10^{-4}.
- *Cube-double protocol and ceiling.* Joint success needs both cube targets set at reset; a configuration that sets only the first is the behavioral negative control. Model-free replay of the logged expert actions gives the protocol ceiling $86.7 \pm 1.2\%$ over three seeds (88/86/86), and the negative control collapses it to 8.0%. Every Cube-double rate in this paper is read against that ceiling.
- *Margin replay, dual-gated.* The gain-threshold curves replay the stored executed actions deterministically and record the scored-object distances after every step. Gate one requires the replayed per-episode successes to equal the stored ones exactly; gate two requires $(\text{margin} \leq 0.04)$ to reproduce the stored successes exactly. Gate two caught the first version of the instrument, which included the pre-step initial state in the margin window although the environment scores only post-step states; with the window corrected to $t \geq 1$, all 54 replay runs pass both gates, and setting the threshold to 0.03 as a negative test trips gate two on 3 of 50 episodes as it should.
- *Probe response convention.* A probe state counts as responsive when the scored objects move by more than 10^{-3} m under blind perturbations at any point over the horizon; the non-responsive Scene states are exactly 0.0 throughout, so this criterion is convention-independent there. The Cube-double count applies the stricter per-direction median convention, median displacement over signs and blocks maximized over blind directions: 6 of 8 states respond, against 7 of 8 under the any-movement criterion. On Scene, 3 of 8 candidate states are excluded fail-closed for button flips before any comparison. Probe amplitudes are conservative, about 0.18 of the blind motion the planner actually executes in dataset-scaler units.

C Complete result tables

Table 2: Action-encoder spectra. Singular values of the first-layer action map $\mathbf{W}$, for four checkpoints released by Maes et al. (2026) and the Scene and Cube-double checkpoints we train (Section 5, Protocol). d is the action-block width and $d'=10$ the encoder width. A checkpoint has exactly blind directions when $d > d'$. Reacher enters only as a spectrum and is not evaluated in closed loop. The spectra of $\mathbf{C}$, the second moment of the cost gradient, are in Figure 3b.

checkpoint	d	rank	exact blind dims	tail σ/σ_1
Cube	25	10	15	$0.56, 0.53, 0.44$
Scene	25	10	15	$0.68, 0.62, 0.49$
Cube-double	25	10	15	$0.59, 0.56, 0.42$
PushT	10	10	0	$0.35, 0.15, 0.01$
Reacher	10	10	0	$0.33, 0.32, 0.16$
Two-Room	10	10	0	$0.34, 0.27, 0.22$

The parent DINO-WM release uses encoder width $d'=10$ across its eight environments, with action blocks that wide or narrower (Zhou et al., 2024). Cube's block stacks five control steps of a 5-dimensional arm command, so it is wider than the encoder (Table 2).

Scene and Cube-double: closed-loop grids and the projection ledger. Per-seed success rates (%, 50 episodes each). Scene: ambient 80/72/80/82/78/72, pruned 78/74/70/74/82/64, projection 82/72/82/74/78/70, random-subspace 80/64/72, null-space 68/54/46. Cube-double: ambient 30/36/28/40/30/32, pruned 24/32/26/48/32/34, projection 28/42/32/46/28/36 then 34/36/24/40/36/34 on the six added seeds, random-subspace 26/38/32, null-space 16/26/16; all Cube-double rates sit under the $86.7 \pm 1.2\%$ replay ceiling. The Cube-double projection ledger, in run order: the first six seeds gave $b_{01}=15$, $b_{10}=7$ ($+2.7$ points, CI $[-0.3, +5.7]$, $p=0.134$); the six preregistered follow-up seeds gave 9/9 ($+0.0$, CI $[-2.7, +2.7]$, $p=1.0$); pooled, 24/16 ($+1.33$, CI $[-0.7, +3.5]$, $p=0.268$). The executed blind component cross-checks per environment: solver instrumentation gives medians 2.89/2.98/2.83 per block (Cube/Scene/Cube-double), an independent

Table 3: Full gain-threshold grid. Paired projection−ambient differences re-scored at each threshold τ, with 95% episode-bootstrap intervals; † marks the deployed threshold.

	Cube (n=300)		Cube-double (n=600)	
τ (m)	Δ (pp)	CI_{95}	Δ (pp)	CI_{95}
0.010	+2.3	$[-1.7, +6.3]$	+0.2	$[-1.3, +1.7]$
0.015	+3.3	$[-0.7, +7.7]$	+0.3	$[-1.5, +2.2]$
0.020	+4.7	$[+0.7, +8.3]$	+0.3	$[-1.7, +2.3]$
0.025	+5.7	$[+2.0, +9.3]$	+0.7	$[-1.2, +2.5]$
0.030	+4.7	$[+1.3, +8.3]$	+2.0	$[+0.2, +3.8]$
0.035	+6.0	$[+2.3, +9.7]$	+1.7	$[-0.2, +3.7]$
0.040†	+6.7	$[+3.3, +10.0]$	+1.3	$[-0.7, +3.3]$
0.045	+5.3	$[+2.0, +8.7]$	+2.0	$[+0.0, +4.0]$
0.050	+4.3	$[+1.3, +7.3]$	+1.8	$[-0.3, +4.0]$
0.055	+4.7	$[+2.0, +7.3]$	+1.5	$[-0.7, +3.7]$
0.060	+4.3	$[+1.7, +7.0]$	+2.2	$[+0.0, +4.5]$
0.065	+3.7	$[+1.3, +6.3]$	+2.3	$[+0.0, +4.7]$
0.070	+3.7	$[+1.3, +6.3]$	+3.0	$[+0.8, +5.3]$
0.075	+3.3	$[+1.0, +6.0]$	+2.7	$[+0.7, +4.8]$
0.080	+3.3	$[+1.0, +6.0]$	+2.5	$[+0.5, +4.7]$
0.085	+3.3	$[+1.0, +6.0]$	+2.0	$[+0.0, +4.0]$
0.090	+3.0	$[+0.3, +5.7]$	+1.7	$[-0.2, +3.7]$
0.095	+3.0	$[+0.3, +5.7]$	+2.7	$[+0.7, +4.7]$
0.100	+3.3	$[+0.7, +6.0]$	+2.2	$[+0.3, +4.0]$

reconstruction of the stored executed commands under the dataset scaler gives $2.87/2.94/2.86$, and the same reconstruction on the pruned arm gives 9.8×10^{-8}, machine zero. Table 3 lists the full gain-threshold grid behind Figure 7.

Cube: dose grid and metric-caliber replication. The left table gives sustained $\mathcal{N}(\mathbf{W})$ injections during closed-loop planning, paired against uninjected episodes, with 5 seeds and 8 episodes per amplitude. Amplitude 0.5 is the standard dose of the growth measurements, and the planner's own blind noise sits at ≈2.9 (Section 5.2). Outcomes are invariant across the plateau, and the injections are delivered in full: execution-domain clip rates stay near the baseline's own 3.5%. The right table repeats the growth measurements of Section 3.2 with perturbation magnitude equalized in the deployment metric instead of isotropically. Conclusions are unchanged.

amplitude	flips b_{01}/b_{10}	p	ΔSR (pp)	clip rate
0.25	0/0	1.00	+0.0	3.5%
0.5	0/0	1.00	+0.0	3.6%
1	0/0	1.00	+0.0	3.8%
2	1/0	1.00	+2.5	4.2%
4	0/3	0.25	−7.5	6.3%

quantity	isotropic	deployment metric
blind slope	0.27 [0.09, 0.48]	0.25 [0.07, 0.46]
visible slope	0.41 [0.17, 0.69]	0.41 [0.17, 0.69]
slope ratio	0.66 [0.51, 0.74]	0.62 [0.41, 0.70]

PushT: rank sweep and surgical-model grid. The left table gives the per-rank paired comparison against the intact model (r=10), ambient sampler, 50 episodes per seed. The r=8, 7 rows are the zero-net-cost region with symmetric flips, and the r=5 row is the only one with a measured loss. The right table runs the cross-environment grid of Section 5.1 on the r=7-truncated model, the controlled-contrast configuration used before the healthy-model protocol was adopted (mean SR over 3 seeds). There the pruned and ambient samplers coincide within noise, and the random-subspace control is consistently worse ($p = 0.009$–0.029).

r	seeds	SR (%)	b_{01}/b_{10}	p	per-seed Δ
10	12	87.2	—	—	—
8	3	88.7	4/4	1.0000	-6,+8,-2
7	3	88.7	6/6	1.0000	+0,+0,+0
5	12	82.0	26/57	0.0009	-16,+0,+0,-2,-2,-4, +4,-12,-4,-10,-4,-12

sampler	N=30	N=100	N=300
ambient	74.0	82.7	88.7
pruned	72.7	85.3	91.3
random	63.3	73.3	79.3

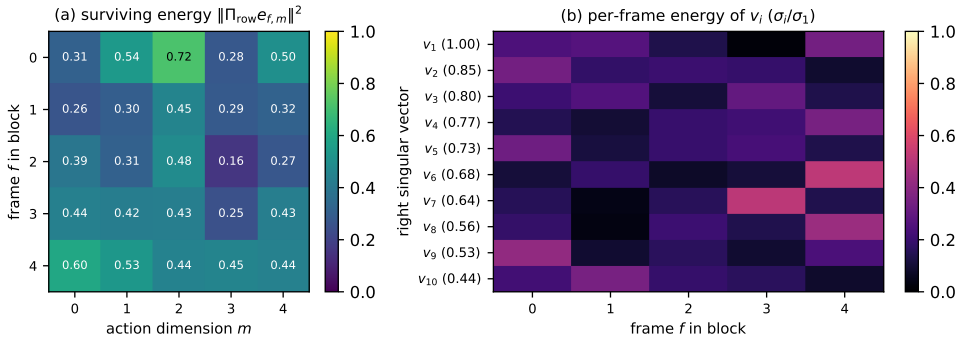


Figure 4: Visible-subspace structure on Cube. (a) Energy of each $(\text{frame}, \text{dimension})$ coordinate surviving projection onto $\text{row}(\mathbf{W})$. **(b)** Per-frame energy of each right singular vector, with σ_i/σ_1 in parentheses.

Executed-command metrics. The table below reads the executed commands themselves (Cube, $N{=}300$, six seeds, 3,000 blocks per arm; medians): the fraction of block energy in $\mathcal{N}(\mathbf{W})$, the per-step command change in standardized and environment units, and the command norm. This pipeline is independent of the solver instrumentation and reproduces its median blind norm (2.87 against 2.89). The ambient sampler spends 40% of executed energy in the blind subspace. Pruned sampling and the execution projection both remove it to numerical zero, roughly halve the per-step command change, and shrink the command norm.

sampler	blind frac.	$\|\Delta a_t\|$ (std)	$\|\Delta a_t\|$ (env)	$\|a_t\|$
ambient	0.40	2.30	0.93	1.89
pruned	0.00	1.15	0.49	1.36
exec. projection	0.00	1.25	0.53	1.48

Structure of the visible subspace. Figure 4 maps the Cube decomposition onto the $(\text{frame}, \text{dimension})$ grid of the action block. No ambient coordinate is fully visible or fully blind (surviving energy 0.16 to 0.72), so the blind subspace cannot be found by inspecting coordinates; it requires the SVD. The frame-marginal energy is highest at the block endpoints (2.34 and 2.46 of rank 10) and lowest mid-block (1.62): the bottleneck preferentially preserves where a block starts and ends and discards mid-block detail.

Imagination overestimates visible-direction divergence. Along *visible* directions the imagined divergence at $H{=}12$ exceeds the measured encoded divergence by $\approx 2.8\times$ (0.70 vs. 0.25 median). A drift bound should therefore not be built on imagined divergence, even where the model is sensitive to the perturbation. Our drift bounds are fit on environment-side data only.

Closed-loop context. Replanning-cycle experiments on Cube give three reference numbers. Blind closed-loop drift is ≈ 0.32 $[0.17, 0.53]$ of the planner's own RNG noise floor ($5/5$ seeds). Reseeding the planner flips outcomes where blind injection flips none ($40/40$ invariant). Feedback absorption is not detectable at this power ($D_{ab}/D_{ol} = 0.73$ $[0.35, 1.39]$, CI containing 1).

D Decoder instrument details

The decoder is a context-conditioned U-Net, with context frames supplying appearance through skip connections and the 192-d latent injected at the bottleneck, trained on 40k latent–frame pairs. Unconditional decoding collapses to a mean arm pose because linearly decodable pose information in the latent is weak (ridge probe $R^2 \approx 0.13$, a lower bound on what a nonlinear readout could recover). This is itself consistent with the compression this paper studies. Every black panel of Figure 5 is audited. Reset-frame pairs decode to exactly-zero difference, and the imagined-difference panels are float-level nonzero (≤ 0.62 of a 765 full scale at $H{=}12$), growing in proportion to the latent gap.

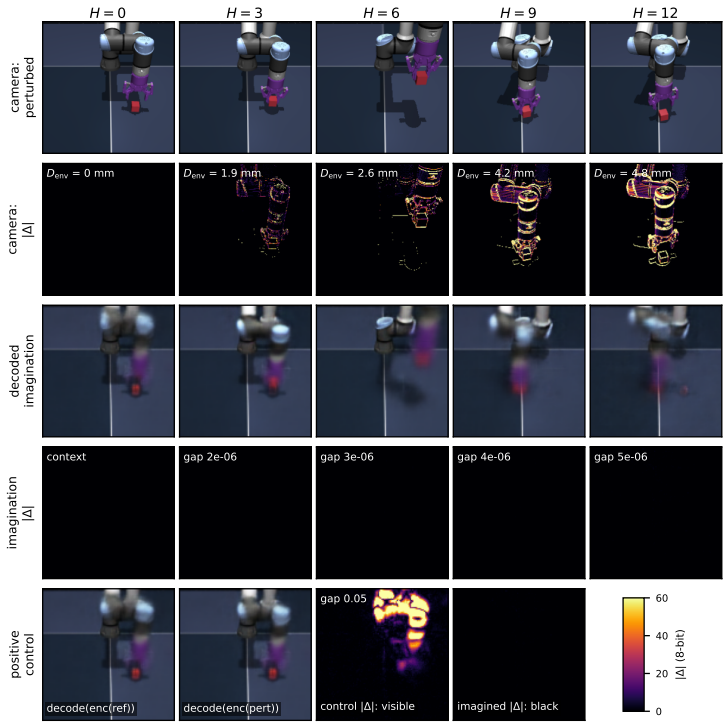


Figure 5: The dissociation rendered to pixels. A context-conditioned decoder, trained post hoc and never used in planning. Row 1: the perturbed rollout seen by the simulator camera. Row 2: the camera-frame difference, which grows. Row 3: the decoded imagined rollout under the same blind perturbation. Row 4: the decoded imagined difference, black at every depth because the imagined latents differ only at the float32 floor. Row 5: the positive control at $H{=}12$, where the same decoder and color scale applied to *encoded real frames* light up along the arm. Black panels are measured values, not rendering artifacts.

E SPECTRAL SURGERY

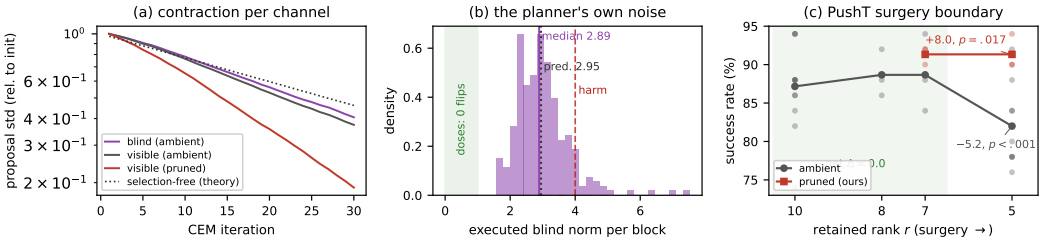


Figure 6: Mechanism and rank sweep, with the surgery panel. Panels **(a)** executed blind norm per block against CEM iteration. **(b)** sampling-scale contraction over the same iterations. **(c)** PushT success under spectral surgery truncating $\mathbf{W}$ to rank r, with 150 paired episodes per rank at $r{=}8, 7$ and 600 at $r{=}5$. Green marks the ranks whose removal leaves aggregate success unchanged.

Spectral surgery deletes the directions the σ-tail heuristic of Section 4.3 prunes, taking them out of the PushT model rather than the sampler (Figure 6c). Truncating the two or three weakest singular directions of the PushT model ($r{=}8$ and $r{=}7$; $\sigma/\sigma_1 \leq 0.35$, the weakest two at or below 0.15) leaves *aggregate* success exactly unchanged over 150 paired episodes per rank: the paired flips are symmetric ($4/4$ and $6/6$), while individual seeds move by up to 8 points in either direction (Appendix C). Trajectories rearrange, but net outcomes do not. Truncating to $r{=}5$ costs -5.2 points

Table 4: Drift-bound widths at matched coverage. Drift bounds over five random state splits (mean±sd, nominal coverage 0.90). Cube is calibrated on sixteen probe states spanning three perturbation amplitudes; PushT on eight magnitude-matched states. All three bounds share the same split-conformal layer, so coverage is aligned by construction and the comparison is width. Bold marks the bound that is strictly narrowest on every one of the five splits in each environment (10/10 combined).

	Cube		PushT	
bound	coverage	width (m)	coverage	width (px)
direction-resolved (ours)	0.86±0.02	**0.0120±0.0071**	0.85±0.08	**12.0±7.5**
global Lipschitz	0.89±0.03	0.1246±0.0039	0.85±0.07	16.1±6.7
scalar power law	0.87±0.03	0.0172±0.0086	0.85±0.07	16.3±7.0

(600 paired episodes, twelve seeds, b_{01}=26 against b_{10}=57, $p = 0.00088$, nine seeds degrading and one improving).

The recovery result reframes what that damage is. Pruned sampling on the damaged r=5 model restricts proposals to the model's own surviving row space. It recovers +8.0 points over its matched ambient baseline (83.3% on the same seeds, $p = 0.017$, all three seeds positive) and reaches 91.3%. Executed noise in the newly blind directions therefore accounts for most of the surgery penalty. This is the mechanism of Section 5.2 at work, and evidence that pruning protects a planner whose model has degraded. The execution projection behaves the same way on the damaged model. It recovers +6.0 points against the matched baseline ($p = 0.064$, two of three seeds positive), and full pruned sampling is statistically indistinguishable from it (+2.0, $p = 0.58$). Pruned sampling and the execution projection coincide here, as they do on Cube. The sampler adds the in-loop form and its guarantees. It does not add a measurable increment.

F Direction-resolved drift bounds

The decomposition also answers a runtime question that the model itself cannot: how far will the environment have drifted from the imagined rollout after H blocks, given the perturbation that the planner is about to execute? On probe rollouts we fit, at each horizon, a linear map from the perturbation's band energies to the observed drift by nonnegative least squares. The band energies are $e = (\|\mathbf{\Pi}_1\,\delta\|, \|\mathbf{\Pi}_2\,\delta\|)$, computed on the visible/blind split for Cube and on the strong/weak singular bands for full-rank PushT. A split-conformal layer Vovk et al. (2005) then turns the point prediction into an upper bound $C(H) = \hat{g}(H; e) + \hat{q}_{1-\alpha}(H)$, where $\hat{q}_{1-\alpha}(H)$ is the empirical $(1-\alpha)$-quantile of calibration residuals at each horizon. With eight to sixteen probe states as the exchangeability unit the finite-sample conformal correction would be vacuous, so we claim empirical rather than certified coverage and report it per calibration split.

The same conformal layer wrapped around two standard aggregates, a single global Lipschitz constant Terver et al. (2026) and a scalar power law, provides coverage-matched baselines: coverage is aligned by construction, so the comparison here is purely about how much tightness direction resolution buys.

The Cube bounds are calibrated on sixteen probe states spanning three perturbation amplitudes, so the scalar baselines receive informative magnitude features. Across five random state splits the direction-resolved bound is the narrowest of the three coverage-matched bounds on every split. Its width sits about 30% below the best scalar baseline, with per-split ratios from 0.37 to 0.88 and mean and median near 0.67. The global-Lipschitz form fits the amplitude-varied data poorly and is an order of magnitude wider.

Empirical coverage on Cube is 0.86 to 0.89 against the 0.90 nominal level, and its split-to-split standard deviation is five times smaller than under eight states. PushT retains the original eight-state, magnitude-matched calibration. There the direction-resolved bound is again narrowest on every split, but the margin depends on the split, so we claim consistency of direction rather than a fixed factor. Magnitude matching also explains why the two PushT baselines nearly coincide: any predictor of the form scalar magnitude times $g(H)$ then conformalizes to the same unconditional per-horizon quantile. The amplitude-varied Cube design removes this degeneracy.

Related certification work. Certified world models bound imagination error with one scalar spectral quantity per model, which eliminates the action direction from the bound Wang (2026a); conformal orbit-valid horizons extend a certificate's domain Wang (2026b); admissibility frameworks qualify simulators by validation level Oefinger et al. (2026); conformalized distance fields calibrate obstacle distances Shin et al. (2026). Our bounds differ on resolution: fed direction-resolved band energies of the perturbation about to be executed, the same conformal layer yields per-direction bounds strictly narrower at matched coverage.

G Composition with a learned sampling prior

Table 5: Composing pruned sampling with a learned sampling prior. Four arms on the PRISM-packaged Cube checkpoint: ambient CEM, pruned CEM, the prior alone, and the prior with pruned sampling. The prior is injected by warm-starting the CEM mean. Three seeds, 50 episodes each, at N=300.

arm (Cube, N=300, 3 seeds)	SR (%)
ambient CEM	62.7 $\pm$5.0
pruned	69.3 $\pm$6.4
PRISM prior	81.3 $\pm$1.2
prior + pruned	82.0 $\pm$2.0

Pruning composes safely with a learned sampling prior. PRISM Wang et al. (2026b) releases a trained prior head for the Cube model. Its packaged world-model weights are bitwise identical to the checkpoint of Section 5.1 (all 303 tensors equal), so the prior head is the new component and the comparison is a within-model composition study, not a replication on a new model. We run a four-arm paired comparison (ambient, pruned, prior, and prior with pruned sampling) with 150 paired episodes over three seeds per arm. The prior is injected by warm-starting the CEM mean, following the convention of its own implementation. On the three shared seeds, the ambient and pruned arms reproduce the corresponding episodes of Section 5.1 bit for bit, an incidental cross-run check of the protocol's determinism.

The trained prior is strong on the model it was trained for (Table 1, a paired difference of $+18.7$ points over its three-seed ambient value of 62.7%, $p < 10^{-4}$). The combination is compatible but not additive: adding pruning on top of the prior changes success by $+0.7$ points ($p = 1.0$). On a model where a learned prior already concentrates the visible search, the marginal value of removing blind noise is not resolvable at this sample size.

H Dose response, cost spectra, and active-subspace sampling

Every experiment in this section was preregistered. The predictions and the disposal rules went into version control before the corresponding closed-loop data existed. Every verdict below is quoted against its preregistered criterion. Per-seed success rates for every arm are in the table below. Per-episode vectors are in the supplementary material.

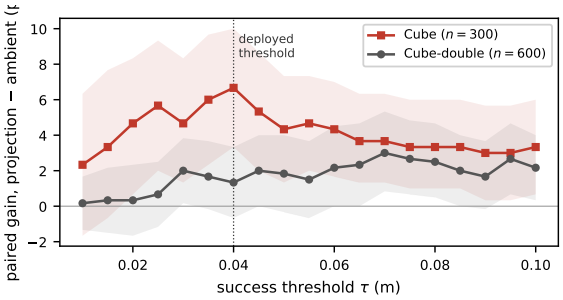


Figure 7: Paired projection gain against the success threshold. Every stored episode is re-scored as a success iff its closest post-step approach (worst block) stays within τ; the deployed threshold is 0.04 m. Bands are 95% episode-bootstrap intervals. The expert-replay ceiling at τ=0.04 is ≈100% on Cube and 90.0% on Cube-double (86.7 ± 1.2 over the three protocol-audit seeds).

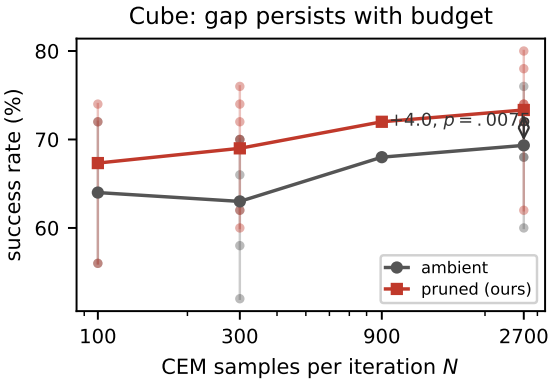


Figure 8: The Cube gap persists over a 27× budget range. Success rate against per-iteration budget N for the ambient sampler and for pruned sampling. Fifty episodes per seed, dots individual seeds, seed counts in Appendix I. The bracket marks the matched-budget paired difference at N=2700.

arm group	per-seed SR (%)
Cube inj σ=1/2/4	68/58/68; 68/54/66; 66/50/54/64/70/50
Scene inj σ=1/2/4	82/74/82; 82/74/80; 74/70/80/80/84/68
Cube-double inj σ=1/2/4	30/38/26; 24/34/26; 28/38/24/40/36/34
Cube active k=4/2/10	76/70/72/72/82/70; 52/44/44; 72/56/72
Cube-double active k=3/1	40/44/38/56/44/48; 26/34/30
PushT / Two-Room active k=2	88/86/84; 86/94/82
Cube MPPI ambient/row/k=4/proj.	48/46/42; 60/52/54; 74/50/72; 56/46/50
Cube-double MPPI ambient/k=3	20/32/20; 42/40/40
Cube MPPI ambient λ=0.1/0.3/1/3	48/40/44; 50/34/42; 50/40/42; 50/48/44
Cube MPPI proj. λ=0.1/0.3/1/3	46/48/50; 56/48/50; 52/52/52; 52/54/52
Cube shooting ambient/k=4/proj.	54/44/48; 66/48/64; 64/52/50
Cube-double shooting ambient/k=3	22/30/26; 40/42/38
Cube iCEM ambient/k=4	54/44/52; 76/62/68
Retrain A ambient/k=4	68/56/70; 74/60/72
Retrain B ambient/k=4/k=3	68/50/66; 74/60/70; 58/46/52
Retrain C (1k eps) ambient/k=4	70/62/66; 72/68/70
Cube-triple ambient/k=3	52/38/44; 48/46/36

Dose-response protocol. The injection arm adds $\delta = \sigma \cdot u$ to every executed block, with u a fresh unit vector in $\mathcal{N}(\mathbf{W})$ per episode and block (generator seeded at 1000+seed), on top of the ambient

Table 6: The same blind subspace, three payoffs. All three manipulation checkpoints erase the same 15 of 25 block directions, execute the predicted blind walk, and collapse under null-space sampling; the payoff of removing the walk differs. Probe response counts states whose scored objects move under blind perturbations; the Cube-double count applies the stricter per-direction median convention (any-movement gives 7/8; Appendix B). Pruned differences use n=300 paired episodes, the Cube-double projection n=600; brackets are 95% episode-bootstrap intervals. Cube-double rates sit under the measured replay ceiling.

	blind walk (pred. 2.95)	probe response	null-space sampler (%)	Δ pruned −ambient (pp)	Δ projection −ambient (pp)	protocol ceiling (%)
Cube	2.89	8/8	63.0→37–45	+6.0 (p=.005)	+6.7 (p=.0002)	≈100
Scene	2.98	1/5	77.3→56.0	−3.7 [−8.0, +0.3]	−1.0 (p=.75)	100.0
Cube-double	2.83	6/8	32.7→19.3	+0.0 [−3.7, +3.7]	+1.3 [−0.7, +3.5]	86.7 ± 1.2

planner's own blind walk; the horizontal axis of Figure 3a is the quadrature total. The zero-dose arm reproduces the ambient arm's per-episode outcomes bitwise, and the k=2/k=1 gap-cut arms double as sensitivity controls. Amplitudes 1 and 2 run three seeds; amplitude 4, the preregistered power point, runs six. Verdicts: the Cube descent at σ=4 is −4.0 with p=0.058 and per-seed signs mixed (+4/−8/−16/−2/0/−2), marginal at six seeds and consistent in direction with the independent grid of Section 5.2; Scene and Cube-double stay under half of Cube's drop, as preregistered.

Spectrum protocol. $\mathbf{C}$ is estimated from 2,000 backpropagated gradients of the frozen cost: 50 evaluation start states × 8 action sequences from the CEM initial proposal × 5 blocks, per checkpoint. The trailing-15 eigenvalue mass is at 10^{-8} of the leading eigenvalue on all three stacked checkpoints, the empirical form of $\mathcal{N}(\mathbf{W}) \subseteq \mathrm{null}(\mathbf{C})$. Bootstrap resampling of the gradients (1,000 draws) moves the chosen subspace by at most 2.6 degrees at the 95th percentile on any gapped checkpoint, and every gap ratio's interval stays clear of 1; Scene's λ_4/λ_5 interval is $[1.08, 1.99]$, so its gapless ruling is stable. Re-estimating under the final-iteration elite distribution of the instrumented runs moves the top-k subspace by at most 6.6 degrees (Cube) and 3.4 degrees (Cube-double), the empirical measure-consistency check quoted in Section 6.

MPPI arms. The planner-transfer arms replace elite refitting with classic full-population softmax weighting at temperature 0.5, the TD-MPC2 convention, fixed globally and never tuned; budgets match CEM exactly (N=300, T=30, equal model calls), and the implementation reproduces the library MPPI solver bit for bit under identity basis (unit test). The ambient MPPI baseline reaches 45.3% on Cube with mean effective sample count $n_{\text{eff}} \approx 1.8$. The weighting analysis of Appendix A predicts executed-noise suppression by averaging over n_{eff} samples, so at $n_{\text{eff}} \approx 1.8$ the execution-projection gain should be comparable to the CEM value, and it is (+5.3 against +6.7). The task tier gains +20.0 on Cube (b_{01}=33, b_{10}=3) and +16.7 on Cube-double (b_{01}=29, b_{10}=4, 40.7% under the 86.7±1.2% ceiling), both $p<10^{-4}$ over three seeds; the visible-subspace tier gains +10.0 (p=0.004). Both directional predictions were preregistered; magnitudes are larger than under CEM, consistent with sampling dilution costing more when the weighting concentrates on few samples, a reading we do not test further here.

Temperature sweep. The grid varies MPPI's softmax concentration over $\lambda \in \{0.1, 0.3, 0.5, 1, 3\}$, ambient and execution-projection arms, three seeds each. Because MPPI never contracts its sampling variance, the blind coordinates of its plan follow a random walk over the T=30 iterations, with per-iteration increment variance $\propto 1/n_{\text{eff}}$. This is why the executed blind norm at λ=0.5 (8.8) exceeds a single raw sample's (3.8), and it predicts that the norm falls as n_{eff} grows. Measured, median n_{eff} rises 1.00 → 7.19 across the grid and the median executed blind norm falls 10.17 → 6.87, strictly monotone (Spearman −1.0), with an endpoint contrast of +3.31 [+3.00, +3.52] over 1,500 blocks per point (preregistered verdict; Figure 9). A displacement instrument, preregistered to flag temperatures where the planned mean degenerates toward zero (median displacement below 0.2× the λ=0.5 reference), excluded no point. The execution-projection gain is positive at every temperature (+4.0 to +9.3, significant at two of five); the preregistered endpoint difference between gains is not resolved (−1.3 [−8.0, +6.0]), so we do not claim the gain tracks the dose: fifty-episode success rates cannot resolve the 1.5× norm change that the per-block instrument resolves.

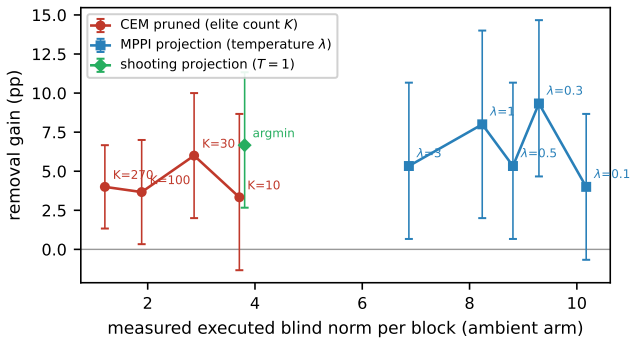


Figure 9: Removal gain against measured executed blind noise, three weighting families. Each ambient arm's executed blind norm (median per block, offline decomposition) against the paired gain of removing that noise: pruned sampling for the CEM elite-count grid, execution projection for the MPPI temperature grid and for shooting. The knobs move the executed noise over an eightfold range; the removal gain stays positive throughout and does not scale proportionally with the dose. Error bars: bootstrap 95% CIs.

Random shooting. Setting $T{=}1$ with a single elite executes the argmin of 300 samples, removing both averaging and iteration. The argmin is independent of the samples' blind coordinates, so the executed blind norm should reproduce a raw draw's χ_{15} statistics exactly: the measured median is 3.808 against the theoretical 3.787, a ratio of 1.006, with the zero-action incumbent selected in 0.3% of blocks (preregistered instrument check; both tolerance gates pass). Under this planner the same restrictions pay again, within matched shooting pairs. The task tier gains $+10.7$ on Cube ($p{=}0.0009$) and $+14.0$ on Cube-double ($p{<}10^{-4}$, mean 40.0% under the $86.7 \pm 1.2\%$ ceiling), and execution projection gains $+6.7$ ($p{=}0.0063$). The budget is one round of 300 model calls, $1/30$ of the iterated planners', so comparisons stay within shooting.

iCEM. The iCEM arms add colored noise ($\beta{=}2$ temporal filtering) and elite retention with EMA refitting (Pinneri et al., 2020), at the library defaults ($\alpha{=}0.1$, five carried elites) and the CEM budget. In the subspace arms the noise is filtered per coordinate of the sampling basis; lifting is a fixed per-timestep linear map, so each ambient coordinate's temporal spectrum is a fixed combination of the coordinate spectra and the low-frequency structure survives. The implementation reproduces the library solver bit for bit under identity basis, with the box clamp both inert and active (unit test). One amendment to the preregistration, made after a crash and before any run completed: the library's clamp is structurally incompatible with this protocol's action blocks (it clamps the 25-dimensional block against per-frame bounds), so both arms run unclamped, like every other solver in this paper. A negative test confirms that disabling the clamp changes the output under tight bounds. Spectral-gap sampling gains $+18.7$ $[+12.0, +25.3]$ ($p{<}10^{-4}$, $b_{01}{=}31$, $b_{10}{=}3$) over the ambient iCEM baseline on Cube.

Task-tier arms. Each k was fixed at the spectral gap before any closed-loop run: 4 on Cube, 3 on Cube-double, 2 on PushT and Two-Room, with $k{=}2$ (Cube) and $k{=}1$ (Cube-double) preregistered as gap-cut negative controls and $k{=}10$ as a span sanity arm (its subspace agrees with $\mathrm{row}(\mathbf{W})$ to 0.02 degrees, as $\mathrm{rank}\,\mathbf{C} = 10$ requires). The Cube-double gain exceeded its preregistered expectation, which predicted an unresolvable increment; per the fixed disposal rule the result was replicated on three fresh seeds before being reported (the replication is stronger than the original batch). The negative controls lose 26.0 (Cube, no episode improved) and 10.7 (Cube-double, $p{=}0.0025$) against their gapped counterparts. Runs cost 220–300 seconds each on one consumer GPU.

The subspace above the gap and action repeat. The action block stores its five control steps one after another, and the action scaler acts on single control steps. A block that repeats one standardized command therefore also repeats one raw command. The action-repeat direction of a command component is the unit vector with entries $1/\sqrt{5}$ at that component of every control step, and the action-repeat subspace is their span. Table 7 gives, for every eigendirection of $\mathbf{C}$ above a gap,

Table 7: Overlap of the subspace above the gap, in full. Shares are per eigendirection of $\mathbf{C}$. Component masses are the squared norms of the projections of the action-repeat directions onto the top-k eigendirections. On the manipulation checkpoints their order is x, y, z, yaw, gripper. The angle columns list all k principal angles in degrees, largest first.

checkpoint	k	gap	share per eigendirection	component mass in top-k	to action repeat	to $\mathbf{W}$ top-k
Cube	4	12.8×	0.999, 0.979, 0.937, 0.952	0.987, 0.999, 0.998, 0.000, 0.883	20.3, 6.1, 1.8, 1.1	79.2, 55.2, 42.9, 13.2
Cube-double	3	13.3×	1.000, 0.999, 0.999	0.998, 1.000, 0.999, 0.000, 0.000	2.5, 2.0, 1.0	82.7, 61.2, 28.4
PushT	2	21.8×	0.981, 0.981	0.977, 0.985	9.1, 6.6	82.2, 46.9
Two-Room	2	42.3×	1.000, 1.000	1.000, 1.000	1.3, 0.7	88.1, 47.8

the share of its squared norm inside the action-repeat subspace. It also gives the squared norm of each action-repeat direction's projection onto the top-k eigendirections, which identifies the kept components. The last two columns are the principal angles from the top-k eigendirections to the kept action-repeat directions and to the top-k right singular vectors of $\mathbf{W}$. The action-repeat directions are not confined to $\mathrm{row}(\mathbf{W})$. The yaw direction has 35% of its squared norm in the blind subspace on Cube and 94% on Cube-double. The gripper direction has 9.9% and 3.4%.

Under the elite measure the fourth Cube eigendirection has 0.856 of its squared norm in the action-repeat subspace.

An action-repeat sampler on these directions ran as a preregistered control, on the kept components and on all five. It samples CEM coefficients exactly as the spectral-gap arms sample theirs, with the same budget, elite count, seeds, and episode draws. Before the control ran, the basis-file code path had to reproduce the archived spectral-gap arm on seed 0. The preregistered criterion asked for bit-identical executed actions. Two runs of the same code differed by 2.1×10^{-7}. Before any control episode, the criterion was relaxed to identical episode outcomes and executed actions equal to within 10^{-5}, and the path met it. The same check on the execution-projection code path did not reproduce its archived run, with a different outcome on one of 50 episodes. Before any PushT episode, the check was redefined as agreement between that path and a rerun of the archived path under the current code, and the two agreed. The action-repeat arms do not use either path. Of the twelve spectral-gap arms paired here, two were re-run under the current code, one per checkpoint, and the rest are archived runs. The predictions, the disposal rule for each of three outcomes, and the analysis script were fixed before any control episode existed. The interval form of this rule and the wording guards of Section 5.3 were drafted before the control started and entered in the ledger after its paired analysis was run. We predicted that the primary interval would contain zero on both checkpoints, and that action repeat on all five components would gain over ambient on both. The first prediction is scored MISS on Cube-double. The second is not scored. The ambient arms are archived runs, and no check reproduced them under the current code. The two comparisons with ambient in Table 8 are therefore descriptive. Table 8 lists every preregistered comparison on Cube and Cube-double. The lower block of the table reports the preregistered PushT arm, a post-hoc projection of executed actions onto the top two eigendirections. We predicted a loss against ambient larger than the 2.0 points of in-loop sampling. This prediction holds.

Full-rank ceilings. Replaying the logged expert actions scores 100% on all three seeds in both full-rank environments, so their flat task-tier outcomes are not an instrument ceiling: roughly twelve points of headroom sit above the 88.0% and 87.3% ambient baselines, and the $k{=}2$ restriction does not capture them. Past the gap, the remaining eight directions carry about 3% (PushT) and 2% (Two-Room) of the gradient mass; we record these fractions as facts rather than as the explanation, because the third-checkpoint pair below shows that post-gap mass does not decide the payoff. The 100% replay also contrasts with the $86.7 \pm 1.2\%$ Cube-double ceiling: the stacking-contact irreproducibility behind that gap is specific to the cube family, and the 2D environments replay losslessly.

A third manipulation checkpoint. We trained a Cube-triple model from scratch (three blocks, joint success; 3,000 oracle episodes, the Cube-double recipe) behind a preregistered gate sequence: its replay ceiling is $89.3 \pm 4.2\%$, so stacking one more block does not widen the contact-replay gap, and its epoch-20 go/no-go passed. Its cost spectrum is the steepest we measure (participation ratio 1.76), with the largest gap at $3|4$ ($6.2\times$, bootstrap CI $[5.4, 7.7]$), so the criterion selects $k{=}3$. The preregistered pairing then scores MISS: -1.3 $[-8.7, +6.0]$ ($p{=}0.86$, seeds mixed) over its own

Table 8: Action-repeat control. Paired differences in points over 300 episode pairs and six seeds, with 95% episode-bootstrap intervals. The other columns give the discordant pairs, the exact McNemar p, and the number of seeds with a positive difference. The lower block is the PushT arm, over 150 pairs and three seeds.

comparison	Δ [CI]	b_{01}/b_{10}	p	seeds > 0
Cube				
spectral-gap − action repeat, kept components (primary)	$+3.0$ $[+0.0, +6.3]$	16/7	0.09	four of six
spectral-gap − action repeat, all five components	$+0.7$ $[-3.3, +4.7]$	19/17	0.87	four of six
action repeat, kept components − ambient	$+7.7$ $[+3.7, +11.7]$	32/9	<0.001	four of six
action repeat, all five components − ambient	$+10.0$ $[+6.0, +14.3]$	36/6	<0.001	all six
action repeat, all five − kept components	$+2.3$ $[-1.3, +6.0]$	19/12	0.28	four of six
spectral-gap − ambient (archived)	$+10.7$ $[+6.7, +15.0]$	39/7	<0.001	all six
Cube-double				
spectral-gap − action repeat, kept components (primary)	-3.3 $[-6.3, -0.3]$	6/16	0.05	two of six
spectral-gap − action repeat, all five components	$+6.7$ $[+1.7, +11.7]$	39/19	0.01	five of six
action repeat, kept components − ambient	$+15.7$ $[+10.7, +20.7]$	58/11	<0.001	all six
action repeat, all five components − ambient	$+5.7$ $[+2.0, +9.7]$	26/9	0.0060	five of six
action repeat, all five − kept components	-10.0 $[-15.3, -4.7]$	18/48	<0.001	none of six
spectral-gap − ambient (archived)	$+12.3$ $[+7.3, +17.3]$	49/12	<0.001	all six
PushT				
post-hoc projection onto top two − ambient	-8.0 $[-13.3, -2.7]$	3/15	0.0075	none of three
post-hoc projection − in-loop sampling, top two	-6.0 $[-12.7, +0.7]$	9/18	0.12	none of three
in-loop sampling, top two − ambient (archived)	-2.0 $[-8.0, +4.0]$	9/12	0.66	one of three

ambient baseline of 44.7%, safe but paying nothing, with forty-five points of headroom under the ceiling. It forms a matched pair with Cube-double: the two checkpoints agree on gap position (3|4), post-gap visible mass (1.4% on both), and near-flat dimension count (seven of ten), yet one gains $+12.3$ and the other nothing. No single spectral statistic we measured decides the dilution payoff, and we scope the dilution-channel claim to Cube-double accordingly.

Retraining the model. Three from-scratch retrains of the Cube world model (two seeds at 3,000 episodes, one at 1,000; the official recipe otherwise) test which spectral facts survive training randomness (Figure 10a). The architectural null space and the leading structure do: every retrain keeps the exact 15-dimensional blind space, and the top two eigendirections stay within 4.3 degrees of the official checkpoint's. The gap parameters do not: the 4|5 ratio is 12.8× on the official checkpoint but 5.5×, 2.7× (largest gap moved to 3|4), and 7.7× across the retrains, non-monotone in data scale; we preregistered gap stability and score it MISS. Closed loop, sampling the top four directions gains $+4.0$, $+6.7$, and $+4.0$ over each retrain's own ambient baseline, positive in nine of nine paired seeds: the payoff of truncating to four directions survives retraining. The criterion's estimate of k does not always survive. On the shallow-gap retrain, the largest gap (3|4 at 4.3×, above the 3× threshold) selects k=3, which loses 9.3 points (p=0.0043, post-hoc probe): with a shallow spectrum the gap position misleads, and the threshold alone does not catch it. We therefore read k off deep gaps only, and the k values in this paper are properties of the released checkpoints.

Out-of-family check (TD-MPC2). The spectrum instrument and the gap criterion port to TD-MPC2 (Hansen et al., 2024) unchanged: J is its own planning objective (discounted reward plus terminal value), ρ its own clamped first-iteration proposal, and states are drawn along greedy evaluation rollouts of the official 5M single-task checkpoints. On the three preregistered dmcontrol tasks, tested in fixed order, every spectrum is gapless: the largest adjacent eigenvalue ratios are 2.56, 1.92, and 1.85 on dog-run, humanoid-walk, and walker-run, all below the preregistered threshold of 3, with participation ratios 6.4/5.1/4.3 against 2.3–3.1 for the manipulation checkpoints. The criterion therefore rules its own truncation out, as it does on Scene, and no truncation arm was run. The subspace planner passes its gates (bit-identical to the official planner under an identity restriction; the determinism check passed after a start-state-drift fix). What transfers out of family is the procedure: the spectrum is measurable and the criterion decides, including deciding not to cut.

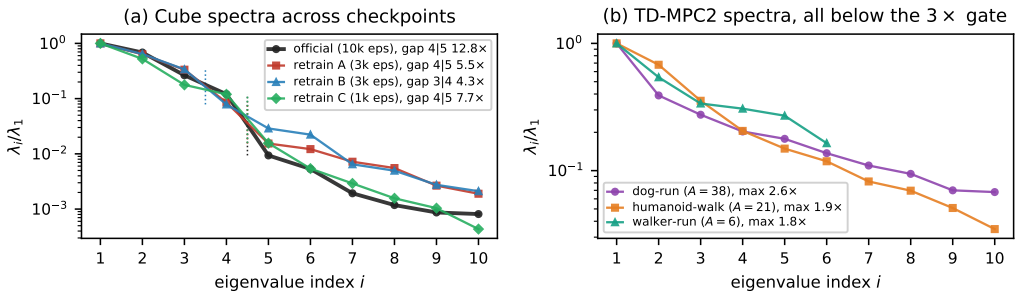


Figure 10: Cost spectra across retrains and across model families. (a) Cube cost spectra for the official checkpoint and three from-scratch retrains: the curves coincide over the leading directions and diverge in the gap region, where each checkpoint's largest gap (dotted) sits at its own position and depth. (b) The three TD-MPC2 locomotion spectra measured by the same instrument, all with largest adjacent ratios below the $3\times$ criterion threshold.

I Reproducibility

Every number in the paper is recomputed from stored per-episode records by the figure and table scripts. No statistic is transcribed by hand. Seed counts: six for the Cube arms at $N{=}300$ and $N{=}2700$ (three at $N{=}100$, one at $N{=}900$), six for the paired Scene samplers, six for Cube-double with twelve for its execution projection, twelve at the surgery endpoints, three elsewhere. The mapping from result to instrument is in Table 9.

Table 9: From result to instrument. Each row names a result of the paper and the code path that produced it. Section and appendix numbers refer to this paper.

result	instrument
floors, determinism (§3.1)	environment self-check suite
growth, dissociation (§3)	open-loop perturbation collector and gate analysis (Fig. 1)
main grids (§5.1, 5.1)	paired grid evaluator (Fig. 8, Tab. 1)
mechanism (§5.2)	instrumented solver and decomposition arms (Fig. 6)
dose grid, closed loop (§5.2)	closed-loop injection harness (App. C)
surgery (App. E)	spectral-surgery evaluator (Fig. 6c)
drift bounds (App. F)	probe collectors and conformal fitter (Tab. 4)
prior combination (App. G)	four-arm composition harness
three-environment boundary (§5.3)	paired grid evaluator and replay-ceiling harness
gain-threshold curves (§5.3)	dual-gated margin-replay instrument
sampler families, retrains, third checkpoint (App. H)	paired grid evaluator with subspace MPPI, iCEM, and shooting solvers
cost spectra, stratified and out-of-family (App. H)	cost-spectrum probe on the frozen models
solver	subspace CEM, MPPI, and iCEM solvers with equivalence unit tests

All experiments run on a single consumer GPU. A standard 50-episode evaluation at $N{=}300$ takes about four minutes on an unshared GPU (249–282 s per Cube-double run across the twelve added arms), the $N{=}2700$ sweep about 26 minutes per batch, and the drift-bound suite is CPU-only. Seeds are enumerated in the per-experiment records. The code, the per-episode records behind every number, and the figure and table scripts are in the supplementary material. The supplementary material is self-contained apart from the checkpoints and datasets, which are hosted externally. It includes a CPU-only self-check that recomputes the paper's statistics from the included records, and it pins the exact environment.