WHAT IN-CONTEXT REINFORCEMENT LEARNERS INFER ABOUT UNSEEN TASKS

Anonymous authors
Paper under double-blind review

ABSTRACT

In-context reinforcement learning agents are often said to generalize to unseen tasks. Yet common protocols either hold out tasks inside the training range, which tests interpolation, or evaluate beyond that range without measuring the inferred task. We instead place evaluation tasks outside the convex hull of the training-task parameters and approximate the agent's implicit task estimate from hidden states, actions, and behavior. On HalfCheetah-Vel and Ant-Dir, decoded estimates flatten near a training boundary. On Ant-Goal, a one-component diagnostic likewise fails to recover the requested outward change. Behavioral readouts also show negative expert-relative shift-axis gaps in six MetaWorld families whose latent estimates cannot be decoded reliably. Classical context encoders show similarly weak requested-direction responses on the two families tested. Routing the prompt through a hand-designed statistic can carry task information beyond the hull. But this construction is not a deployable method, and behavior need not improve when a decoded response moves farther outward. Under explicit convex-readout, bounded-residual, fixed-conditioning, and absolute-suboptimality assumptions, we prove a return-loss floor; proportional outward slack yields a lower bound linear in hull distance. These results help separate failures of task inference from failures of policy execution.

1 INTRODUCTION

In-context reinforcement learning agents adapt to a new task from a few trajectories without gradient updates, and several studies report generalization to held-out tasks (Xu et al., 2022; Wang et al., 2024; Wu et al., 2025; Lee et al., 2023; Polubarov et al., 2026). The meaning of that claim depends on how the held-out set is built. A common protocol samples held-out tasks from the range covered by the training tasks. Prompt-DT states this convention directly: held-out tasks "have goals (target velocity or target direction) within the goal range calibrated by training tasks" (Xu et al., 2022, §6.4); Wang et al. (2024, App. G) repeat the same description before presenting their beyond-range test. In the train/test-context terminology surveyed by Kirk et al. (2023), these are unseen instances but parameter-space interpolation. Controlled function-learning studies likewise show that in-context transformers can select a function from the pretraining family rather than learn a new one (Yadlowsky et al., 2023; Wang et al., 2025); in controlled linear regression, broader task-space behavior can emerge as training diversity increases across a regime-dependent transition (Raventós et al., 2023; Goddard et al., 2025).

We instead build evaluation tasks outside the training range. We train prompt-conditioned decision transformers (Chen et al., 2021; Xu et al., 2022) and a small decision-pretrained transformer (Lee et al., 2023) on offline data from fourteen MuJoCo locomotion (Todorov et al., 2012) and MetaWorld manipulation (Yu et al., 2020) families. Each family varies a physical task parameter. We then estimate the implicit task belief (Zintgraf et al., 2020) in three ways with different failure modes: ridge probes fit only on training tasks and normally use rollout states, inversion of the emitted actions against per-task experts (Haarnoja et al., 2018), and direct measurement of behavior. The one Ant-Goal probe that uses offline expert segments is explicitly marked as contaminated.

On HalfCheetah-Vel, all three measurements show saturation rather than unstable failure. The estimate tracks the target inside the training range and then changes little beyond it. With training speeds in $[1.1, 3.0]$ m/s and a far-band mean target of 3.8 m/s, Prompt-DT's action-based estimate

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

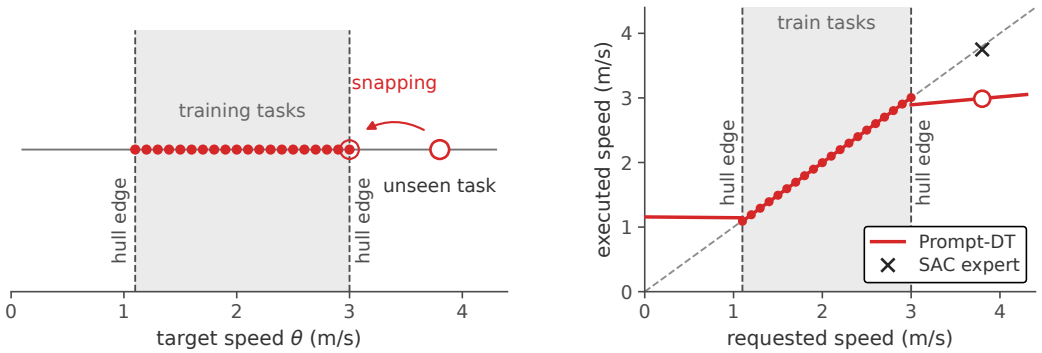

Figure 1: The test on HalfCheetah-Vel. Training speeds lie in the shaded band. When Prompt-DT is asked for a speed above that band, both the decoded estimate and the achieved speed remain near the upper training boundary.

is 2.986 ± 0.023 m/s. Its achieved speed has a slope of only $+0.123 \pm 0.021$ above the hull, while the SAC expert runs at about 3.75 m/s on the same band (Figure 1). The direct speed measurement does not use a decoder. The other two readable families also fail to recover the requested outward change, although Ant-Goal is measured only by the nearest-task component q. Explicit codes from three classical context encoders show weak requested-direction responses as well (Gao et al., 2023; Li et al., 2024; Nakhaeinezhadfard et al., 2025). We do not make latent-estimate claims for the other eleven families because neither decoder passes there.

The HalfCheetah-Vel context itself is informative: a model-free statistic recovers the target about as well outside the range as inside. This differs from MDP ambiguity, in which offline data do not identify the task (Dorfman et al., 2021). We then train an architecture variant that blocks cross-episode attention and routes a hand-designed context statistic through an affine code (Zaheer et al., 2017). The variant shares the family, data, trunk dimensions, and training budget with the control, but it changes the conditioning modules and is trained separately. It therefore tests whether an explicit statistic channel can help; it is not a surgical or causal readout replacement. Among the five MetaWorld families retained by the corrected post hoc screen, mean OOD success is higher in all five and mean in-range success is lower in four; this selected comparison is exploratory.

Moving a decoded response outward does not always improve behavior. On HalfCheetah-Vel below the training range, an oracle-input condition closes about three fifths of the return gap to the expert. Above the range, the same condition performs worse than a nearest-boundary-input condition. Extending the training distribution with exactly relabeled tasks, a controlled analog of virtual-task methods (Lee & Chung, 2021; Kim et al., 2025), moves the saturation level toward the new edge but does not restore a unit out-of-hull slope. Our theory gives a conditional explanation rather than a fitted model of these experiments (Tsai et al., 2019; Yan et al., 2026b). If a task-parameter readout is convex over the training parameters, it cannot leave their hull. With additional slack and return-sharpness assumptions, this yields a return-loss lower bound (Theorem 1). Separately, a population affine readout of a suitable statistic has an error bound independent of hull distance (Theorem 2).

2 Related Work

In-context agents and the encoders they replaced. Decision transformers model offline trajectories as sequences (Chen et al., 2021; Janner et al., 2021). In-context agents add a task-information channel: a trajectory prompt (Xu et al., 2022), a world-model code (Wang et al., 2024), mixture-of-experts routing (Wu et al., 2025), an across-episode learning history (Laskin et al., 2023), or an in-context dataset used to predict expert actions (Lee et al., 2023; Polubarov et al., 2026). Main evaluations often sample held-out parameters from the training range. For example, Prompt-DT interleaves test velocity indices with training indices, while MetaWorld's ML1 protocol samples goal positions from one task-specific randomization (Yu et al., 2020). Some studies also test beyond-range or otherwise shifted tasks (Wang et al., 2024; Wu et al., 2025; Polubarov et al., 2026), but they

report return or action-belief concentration rather than the inferred task parameter. Those measurements cannot rule out a policy that acts on a nearby training task.

Earlier meta-RL methods encode a context into an explicit task variable and condition the policy on it (Rakelly et al., 2019; Zintgraf et al., 2020). Offline variants train this code with a metric loss (Li et al., 2021), behavior-invariant contrastive learning (Yuan & Lu, 2022), context-shift correction (Gao et al., 2023), entropy regularization (Nakhaeinezhadfard et al., 2025), or world-model-based objectives, including multi-step latent temporal consistency (Nakhaei et al., 2026; Qian et al., 2026). Li et al. (2024) show that FOCAL, CORRO, and CSRO optimize bounds on the same task-representation mutual information; ER-TRL controls a related but distinct information term. We therefore compare representative explicit encoders as one model class while reporting each method separately. MACAW instead adapts model parameters through gradient updates rather than using a fixed context encoder, so it lies outside our claim (Mitchell et al., 2021). We are not aware of prior work that plots an explicit offline meta-RL task estimate against the true parameter beyond the training range.

In-context learning outside the pretraining distribution. Supervised work reports error curves across function classes and task distributions (Yadlowsky et al., 2023; Wang et al., 2025; Raventós et al., 2023; Goddard et al., 2025). In sequential decision making, prior work asks which training distributions produce in-context learning and evaluates return or action accuracy (Raparthy et al., 2024). These metrics do not show where a failed task estimate lands, and a direct action-prediction head does not separate inference from execution. Moreover, an out-of-distribution label need not imply an out-of-hull test: most OOD tasks in Kim et al. (2025, §5.1) interpolate between training regions.

Transfer bounds and convex aggregation. Our theoretical starting point is Yan et al. (2026b), who study supervised set-to-scalar regression and show a capacity limit for convex aggregation, including softmax attention. We use the elementary convex-hull consequence but study a different object: the hull of training-task parameters across episodes, not state-action pairs (Yao et al., 2025) or token embeddings within one sample. We explicitly assume that the agent's cross-episode readout acts as a convex smoother over task parameters. Attention can be viewed as a kernel smoother over value vectors (Tsai et al., 2019), but the step from value vectors to task parameters is an idealization. In controlled synthetic transformers, empirical work on task-vector geometry offers partial support (Yan et al., 2026a), while other analyses give different reductions of softmax attention (Xie et al., 2026). With additional bounded-residual, fixed-conditioning, and absolute-suboptimality assumptions, we derive the clipped floor $c \max\{0, d(\theta) - \delta_{\text{out}}(\theta)\}$; proportional outward slack yields a lower bound linear in hull distance. It complements the upper transfer bound of Barreto et al. (2017), which includes distance to a previously solved task under shared dynamics and linear rewards.

3 Putting the evaluation task outside the training hull

We first define the hull precisely. A task family contains MDPs with shared state and action spaces and a parameter $\theta \in \Theta \subseteq \mathbb{R}^d$. Inspired by hidden-parameter MDPs (Doshi-Velez & Konidaris, 2016), we allow θ to affect either dynamics or reward. It may be a target speed, heading, goal position, mass multiplier, or friction multiplier. We write $\hat{\theta}$ for the task estimate on which an adapting agent appears to act. The estimate is explicit in a context encoder (Zintgraf et al., 2020) and implicit in a decision transformer. The training hull is $H = \text{conv}\{\theta_1, \ldots, \theta_N\}$, and $d(\theta) = \min_{u \in H} \|\theta - u\|$ is the distance from an evaluation task to that hull.

Following the standard task-distribution framing of meta-RL (Beck et al., 2025), each family uses a connected middle design region for training and held-out bands beyond its edges; only θ changes within a family. We extend the train/test-context terminology surveyed by Kirk et al. (2023) by reporting each task's hull position. Using one reported direction for most families makes saturation easy to read; Ant-Goal instead uses the task-specific component q in its two-dimensional goal space. Eight families use MuJoCo locomotion (Todorov et al., 2012): target speed, heading, planar goal, mass, or friction. Six use MetaWorld manipulation (Yu et al., 2020): task identity is fixed while the goal moves beyond a nominal training box. Their actual finite-task hull is the convex polytope spanned by the 20 sampled training goals, not the whole box. HalfCheetah-Vel trains on $[1.1, 3.0]$ m/s, with near OOD bands at $[0.6, 1.0]$ and $[3.1, 3.5]$ and far bands at $[0.1, 0.5]$ and $[3.6, 4.0]$. We call hull tasks ID and tasks beyond it OOD. For circular heading tasks we interpret only the near

band to avoid angular wraparound; for MetaWorld we use one OOD band. Appendix B gives every split and the distance to the actual hull.

Each family contains 40 tasks, including $N = 20$ training tasks. Every task has a fixed offline dataset collected during a separate SAC training run, plus the converged SAC expert from that run (Haarnoja et al., 2018; Fu et al., 2020). We evaluate Prompt-DT (Xu et al., 2022), which gives a decision transformer (Chen et al., 2021) a five-step expert prompt, and mini-DPT (Lee et al., 2023), which reads an unordered set of expert transitions. The main grid uses six seeds and five evaluation episodes per task. Before interpreting an agent's estimate, we require its training-task return to reach 0.80 of the expert reference under a floor-normalized score. On HalfCheetah-Vel, Prompt-DT passes at 0.898 ± 0.009 over the three seeds used for that eligibility test. Meta-DT (Wang et al., 2024) fails this test and therefore appears only as a case study in Appendix G.

4 Three ways to read the implicit task estimate

Because the models do not output θ directly, we use three measurements with different weaknesses. **Ridge probes** are maps $\hat{\theta} = Wh + b$ from hidden states, normally recorded while the agent acts; one marked Ant-Goal diagnostic instead uses offline segments. They are fit only on training tasks; layer and regularization are selected by task-level cross-validation (Gurnee & Tegmark, 2024). We keep the probe linear, report an input-layer control, and follow the caution that probe accuracy alone does not establish representation content (Hewitt & Liang, 2019). Related work probes latent world states and task vectors (Li et al., 2023; Hendel et al., 2023). **Action inversion** finds the per-task expert whose action best matches the model's action. It can fail when nearby experts are hard to distinguish or when the policy barely changes. **Behavioral readout** directly measures achieved speed, heading, or position. It needs no fitted decoder, but the chosen physical statistic may still be noisy. In MetaWorld, even expert readouts track the goal with slopes of only 0.607 to 0.898, so we report signed gaps from the expert in meters. Soccer is noisier still because the ball can roll back out of the goal.

Except for the explicitly marked Ant-Goal offline probe, all instruments read the agent while it is acting. The Ant-Goal offline segments already contain expert behavior: the layer-0 task-level cross-validated R^2 is 0.647, and the selected headline layer reaches 0.920. This probe may therefore read behavior already present in its inputs. We treat it as contaminated, not as an estimate of the online code. We use agent-generated states for every other probe and print the layer-0 probe of the same rollouts beside every headline probe.

We report an in-hull slope, an OOD slope, and the mean estimate in each band, with standard deviation across seeds. A hard snap has an in-hull slope near one, an OOD slope near zero, and a band mean near the hull edge. Wide variation without a stable landing point instead suggests irregular failure. For Ant-Goal, let θ_{nn} be the ℓ_1-nearest training task and $u = (\theta - \theta_{\mathrm{nn}})/\|\theta - \theta_{\mathrm{nn}}\|_2$. We report the normalized nearest-task directional component $q = u^\top(\hat{\theta} - \theta_{\mathrm{nn}})/\|\theta - \theta_{\mathrm{nn}}\|_2$: $q = 0$ means no recovered component along this displacement and $q = 1$ means recovery of the true component. The value is unbounded and does not determine hull membership or full-vector recovery. MetaWorld uses expert-relative gaps and axis-only gains relative to the nominal generator-box edge; neither is a hull-distance measure. We fixed two decoder thresholds before evaluation: inversion top-1 accuracy of at least 0.70 and probe cross-validated R^2 of at least 0.70 on training tasks. Appendix D gives the full protocol.

5 Where the estimate lands when the task leaves the hull

Only three of the fourteen families have at least one decoder that passes its training-task quality threshold. HalfCheetah-Vel passes both thresholds for both agents: inversion accuracy is 0.909 and 0.933, and probe R^2 is 0.961 and 0.919. On Ant-Dir, the probes pass at 0.829 and 0.727, while inversion narrowly misses at 0.661 and 0.692. On Ant-Goal, the contaminated Prompt-DT probe passes at 0.920 but its inversion misses at 0.559; mini-DPT has the reverse pattern, with inversion at 0.908 and probe R^2 at 0.607. Table 1 uses action inversion as one common measurement. On Ant-Dir it is read with the passing probe for each agent. On Ant-Goal, Prompt-DT inversion is read only with its passing but contaminated probe, whereas the mini-DPT row rests on its passing

Table 1: **Action-based responses undershoot the requested outward change.** Results are over six seeds. HalfCheetah-Vel action inversion passes its quality threshold. On Ant-Dir, marked †, inversion misses for both agents and is read with their passing probes. On Ant-Goal, also marked †, Prompt-DT inversion is read with its passing but contaminated probe, while mini-DPT inversion itself passes. HalfCheetah-Vel and Ant-Dir response columns are slopes. For Ant-Goal, the in-hull entry is a multivariate OLS slope, the band entries are mean nearest-task components q, lower/upper means moderate/extreme, and the last two columns are mean decoded radii; neither q nor radius determines hull membership.

	response quantity			band-mean readout	
Agent	in hull	lower band	upper band	lower	upper
HalfCheetah-Vel, hull $[1.1, 3.0]$ m/s, bands $+0.300$ and $+3.800$					
Prompt-DT	$+1.008 \pm 0.004$	$+0.000 \pm 0.066$	$+0.151 \pm 0.031$	$+1.128 \pm 0.059$	$+2.986 \pm 0.023$
mini-DPT	$+1.021 \pm 0.007$	$+0.498 \pm 0.158$	$+0.447 \pm 0.035$	$+0.668 \pm 0.098$	$+3.369 \pm 0.040$
Ant-Dir†, sector-centered angle (center 3.063 rad), hull ± 1.492 rad, band means ± 1.963					
Prompt-DT	$+0.981 \pm 0.012$	$+0.374 \pm 0.087$	-0.159 ± 0.096	-1.497 ± 0.033	$+1.197 \pm 0.037$
mini-DPT	$+0.991 \pm 0.005$	$+0.266 \pm 0.108$	$+0.062 \pm 0.015$	-1.572 ± 0.023	$+1.477 \pm 0.044$
Ant-Goal†, outer training radius 1.2, both bands at $r = 1.600$					
Prompt-DT	$+0.816 \pm 0.048$	-0.457 ± 0.066	-0.337 ± 0.072	$+1.032 \pm 0.026$	$+0.973 \pm 0.019$
mini-DPT	$+0.962 \pm 0.020$	-0.273 ± 0.059	-0.166 ± 0.055	$+1.099 \pm 0.024$	$+1.096 \pm 0.041$

inversion rather than its failing probe. The other eleven families are instrument-limited, so we make no latent-estimate claim for them. Appendices D and F report every threshold and control.

Saturation of the decoded estimate. The tabulated in-hull action-inversion response slopes range from $+0.816 \pm 0.048$ to $+1.021 \pm 0.007$ across the three families in Table 1. Standalone inversion interpretation is limited to HalfCheetah-Vel and mini-DPT on Ant-Goal; the remaining rows require the qualified companion probe described above. On HalfCheetah-Vel, Prompt-DT's slope falls to $+0.151 \pm 0.031$ above the hull and $+0.000 \pm 0.066$ below it. A true upper-band mean of 3.800 m/s decodes to 2.986 ± 0.023 m/s near the upper edge of 3.0; a lower-band mean of 0.300 decodes to 1.128 ± 0.059 near the lower edge of 1.1. The estimate therefore stops systematically rather than scattering across seeds. On this family, action inversion, the rollout-state probe, and achieved speed agree qualitatively (Figure 2). The layer-0 slope also collapses from $+0.909 \pm 0.003$ inside to $+0.079 \pm 0.017$ above the hull, so the probe alone does not identify attention as the cause.

Architecture changes the degree of extrapolation without reaching the requested task. mini-DPT reaches 3.369 ± 0.040 m/s on the HalfCheetah-Vel far band, beyond the old edge but below the requested mean of 3.800. On Ant-Dir, Prompt-DT's upper-band mean remains at 1.197 ± 0.037 rad, inside the edge at 1.492. On Ant-Goal, action inversion gives a negative nearest-task component q for Prompt-DT, -0.457 ± 0.066, so the band-mean decoded component points opposite the requested nearest-task displacement. Because inversion misses its line and the companion passing probe uses offline expert segments, this observation is descriptive and does not establish the deployed latent estimate.

Behavior at the same boundary. The SAC reference changes only modestly with distance on HalfCheetah-Vel, from returns of -22.1 to -25.7 and -31.9 across the three splits. Prompt-DT falls much more, from -27.63 ± 0.73 on training tasks to -91.51 ± 2.20 and -185.07 ± 5.95. The model's degradation therefore cannot be explained by the change in expert difficulty alone. Directly measured speed follows the target with slope 1.008 inside the hull and only $+0.123 \pm 0.021$ above it.

No MetaWorld decoder passes its threshold, so the evidence there is behavioral: success and the signed difference between the agent and expert readouts along the shifted goal axis. The SAC expert succeeds on every task, yet Prompt-DT's soccer success falls from 0.925 ± 0.063 in range to 0.217 ± 0.121 OOD (Table 2). All twelve agent-by-family OOD gaps are negative, from -0.015 to -0.155 m, so each agent stops short of the expert in the direction of the training region. On training tasks the same readouts are within 0.023 m of the expert. For reach, Prompt-DT's OOD readout gap is -0.0473 ± 0.0008 m; separately, the held-out target lies 0.0366 m beyond the nominal box edge on average. Figure 2 shows all twelve gaps.

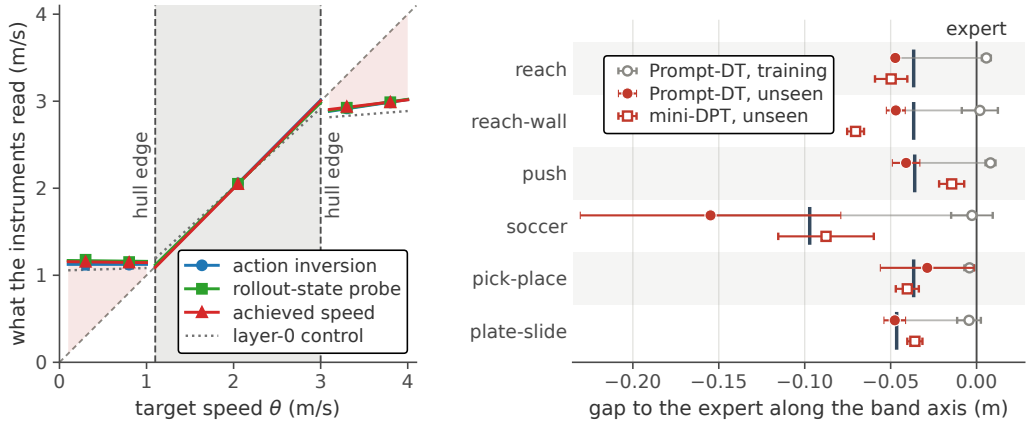

Figure 2: The same HalfCheetah-Vel rollouts read three ways: by inverting the emitted actions against the per-task experts, by a ridge probe on rollout states fit on training tasks only, and by the forward speed the robot settles at. On the right, six MetaWorld families, where the evidence is how far short of the expert the agent stops.

Table 2: **Where no decoder passes, behavior lags the SAC expert along the OOD shift axis.** Results are over six seeds. The last column is the mean target-axis displacement beyond the nominal generator box, a design reference rather than a behavioral gap or a hull distance; *n/a* marks values unavailable from the logged rollouts.

	Prompt-DT success		mini-DPT success		OOD gap (m)		
Family	training	OOD	training	OOD	Prompt-DT	mini-DPT	nominal-box target excess (m)
reach	0.975±0.038	0.600±0.082	*n/a*	*n/a*	−0.0473±0.0008	−0.0497±0.0094	0.0366
reach-wall	*n/a*	*n/a*	*n/a*	*n/a*	−0.0470±0.0055	−0.0704±0.0049	0.0366
push	0.967±0.037	0.550±0.096	0.933±0.037	0.633±0.125	−0.0410±0.0080	−0.0145±0.0073	0.0360
soccer	0.925±0.063	0.217±0.121	0.875±0.063	0.317±0.107	−0.1548±0.0758	−0.0876±0.0278	0.0971
pick-place	0.992±0.019	0.633±0.170	0.983±0.024	0.733±0.075	−0.0287±0.0273	−0.0403±0.0067	0.0366
plate-slide	1.000±0.000	0.717±0.069	1.000±0.000	0.867±0.047	−0.0476±0.0063	−0.0359±0.0044	0.0465

The information already in the context. On HalfCheetah-Vel, a model-free audit rules out an uninformative prompt. Regressing mean prompt speed on the target gives slopes of $+1.008$, $+1.007$, and $+1.043$ below, inside, and above the hull. This five-step trajectory prompt carries the parameter well but yields the smallest decoded OOD response among the compared arms. By contrast, a 100-transition i.i.d. prompt has an audit slope of $+0.987$ below the hull and an action-inversion slope of $+0.427 \pm 0.085$ over three seeds. Thus, on this family, the saturation depends on how the model uses the context rather than on whether the context contains target-speed information.

Explicit task codes in classical encoders. We also test CSRO (Gao et al., 2023), UNICORN (Li et al., 2024), and ER-TRL (Nakhaeinezhadfard et al., 2025) on HalfCheetah-Vel and Ant-Goal, with three seeds each. A ridge decoder fit only on training tasks reads their explicit codes. On HalfCheetah-Vel, in-hull slopes are 0.934 ± 0.058, 0.998 ± 0.000, and 0.970 ± 0.008; beyond-hull slopes fall to 0.067 ± 0.005, 0.098 ± 0.039, and 0.054 ± 0.021. Ant-Goal gains are near zero or negative. The OOD codes also lie close to the outermost training-code clusters. These observations are consistent with weak requested-direction responses but do not rule out nonlinear encoding or decoder extrapolation error.

6 Changing the Path from the Prompt to the Policy

The HalfCheetah-Vel audit shows that its context carries the parameter even when the estimate stops. We therefore rebuild the conditioning path while holding the family, dataset, trunk dimensions, and 100k-iteration budget fixed. The control is Prompt-DT, whose prompt tokens are processed by learned attention. The statsC variant blocks cross-episode attention and sends the prompt to the trunk through $z = W\psi + b$, where ψ is a family-specific statistic and z enters through feature-

Table 3: **Changing the prompt-conditioning path can increase the reported response.** Results cover three families over six seeds. Bold marks the larger reported response quantity; for Ant-Goal this concerns only one displacement component, not full two-dimensional tracking. Layer-0 rows are controls. Ant-Goal uses a contaminated offline probe and is interpreted only as suggestive.

	lower side			upper side		
Instrument	control	filmB	statsC	control	filmB	statsC
HalfCheetah-Vel, beyond-hull slope, below the hull / above the hull						
action inversion	$+0.000 \pm 0.066$	$+0.262 \pm 0.179$	$\mathbf{+0.566 \pm 0.277}$	$+0.151 \pm 0.031$	$+0.408 \pm 0.182$	-3.607 ± 0.934
rollout-state probe	-0.011 ± 0.054	$+0.294 \pm 0.151$	$+0.580 \pm 0.164$	$+0.147 \pm 0.043$	$+0.454 \pm 0.095$	$+0.933 \pm 0.169$
layer-0 control	$+0.029 \pm 0.064$	$+0.102 \pm 0.087$	-0.127 ± 0.434	$+0.079 \pm 0.017$	$+0.238 \pm 0.105$	-2.462 ± 0.553
achieved speed	-0.012 ± 0.066	$+0.273 \pm 0.260$	$\mathbf{+1.296 \pm 0.155}$	$+0.123 \pm 0.021$	$+0.430 \pm 0.161$	-3.930 ± 0.963
Ant-Dir, beyond-sector slope, below the sector / above the sector, moderate band						
action inversion	$+0.374 \pm 0.087$	*n/a*	$+0.428 \pm 0.083$	-0.159 ± 0.096	*n/a*	$+0.388 \pm 0.141$
rollout-state probe	$+0.357 \pm 0.103$	*n/a*	$+0.310 \pm 0.088$	-0.147 ± 0.041	*n/a*	$+0.741 \pm 0.120$
layer-0 control	$+0.373 \pm 0.082$	*n/a*	$+0.353 \pm 0.200$	$+0.036 \pm 0.113$	*n/a*	-0.097 ± 0.237
achieved heading	$+0.333 \pm 0.207$	*n/a*	$+0.126 \pm 0.401$	$+0.040 \pm 0.050$	*n/a*	$+0.660 \pm 0.340$
Ant-Goal, nearest-task component q, moderate / extreme; $\pm$ = task-mean of per-task seed SD						
action inversion	-0.457 ± 0.188	*n/a*	-0.336 ± 0.251	-0.337 ± 0.189	*n/a*	-0.171 ± 0.237
offline probe	$+0.139 \pm 0.156$	*n/a*	$+0.967 \pm 0.042$	-0.143 ± 0.123	*n/a*	$+0.977 \pm 0.053$

wise modulation (Zaheer et al., 2017; Perez et al., 2018). The statistic is mean forward speed on HalfCheetah-Vel and a selected-observation-coordinate summary on MetaWorld. A third variant, filmB, used only on HalfCheetah-Vel, keeps the bottleneck but learns the aggregation. The variants are trained separately and have different conditioning modules, so this is an architecture intervention rather than a readout-only causal test. It shows what can be carried through a supplied statistic; it does not show that attention alone caused the original pattern.

Locomotion. Below the HalfCheetah-Vel hull the control is flat under all three instruments while statsC reads $+0.580 \pm 0.164$ on the probe and $+1.296 \pm 0.155$ on achieved speed. Therefore the robot really does run slower when it is asked for a slower speed, while the layer-0 control does not move with it, at -0.127 ± 0.434 (Table 3). Above the hull the two readings part company: the rollout-state probe has slope $+0.933 \pm 0.169$, while achieved speed runs the other way at -3.930 ± 0.963 and the far-band return falls to -345.3 ± 61.6 against the control's -185.1 ± 6.0. Note that we never read a statsC inversion slope on its own, since the arm lowers the training identification rate of inversion in every MuJoCo family, 0.909 to 0.561 here, so its -3.607 ± 0.934 above the hull reports a halted policy, not a collapsed code.

Ant-Dir changes on one side only: above the training sector, the probe slope goes from -0.147 ± 0.041 to $+0.741 \pm 0.120$ and the achieved-heading slope from $+0.040 \pm 0.050$ to $+0.660 \pm 0.340$. Below it, the two variants lie within each other's seed variation, where the control already moved outward. The upper-band probe mean crosses the sector edge of 1.492 rad, reaching 1.728 ± 0.050 instead of the control's 1.196 ± 0.056. On Ant-Goal, the offline-probe q rises from $+0.139 \pm 0.156$ to $+0.967 \pm 0.042$, while action inversion remains negative and return is slightly worse. The fitted estimate in a separate corrected retrospective pre-check crosses the selected supporting plane on only five of ten moderate tasks. The probe is also contaminated by using offline expert segments, and only contaminated online code probes are available. The contrast is therefore suggestive, not evidence that the deployed policy received the true goal.

Manipulation. Where no decoder clears its threshold, we use behavior only. In the five retrospectively retained families, mean OOD success rises from 0.563 to 0.740, and the signed gap to the expert along the band axis shrinks in each one. On pick-place it becomes $+0.004 \pm 0.039$ m, which is not distinguishable from zero (Table 15). Training-task success falls in four of the five, by 0.19 on soccer. These selected-family averages show better OOD behavior together with lower in-range accuracy; no variant clearly exceeds the SAC expert on the band-axis statistic. On HalfCheetah-Vel, filmB's probe lies between the control and statsC, at $+0.294 \pm 0.151$ below and $+0.454 \pm 0.095$ above the hull. Because the variants are trained separately, this ordering does not isolate which architectural change produced the difference.

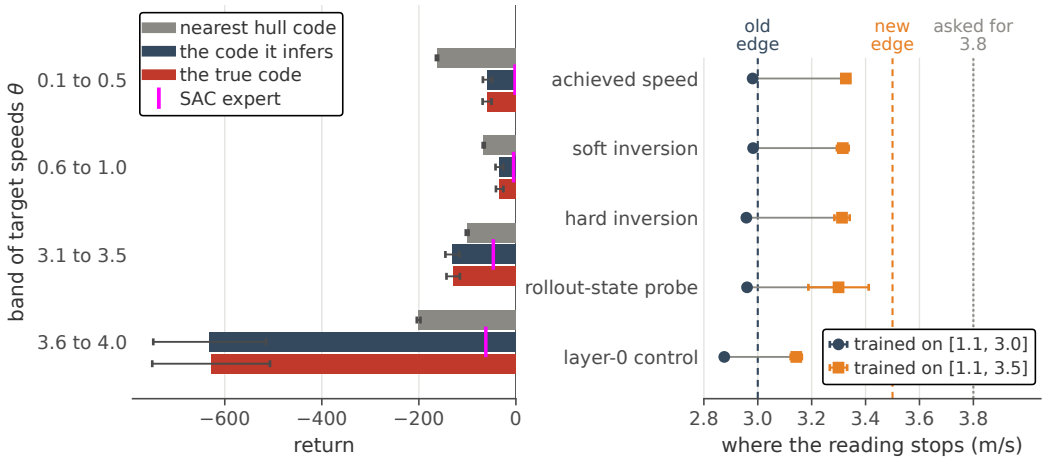

Figure 3: Two intervention results on HalfCheetah-Vel. Left: return when the frozen policy receives the nearest-boundary, measured-statistic, or oracle-input code. Right: far-band readouts after the training range is widened to 3.5.

7 Failures of inference and failures of execution

On HalfCheetah-Vel, the statsC checkpoint admits a controlled oracle input: replacing its measured speed statistic by the requested target and applying the checkpoint's own affine map gives the code associated with that target. This substitution is exact, although the measured statistic has only been shown empirically to track the target rather than proved unbiased. The measured-statistic and oracle-input returns are close in every band, so we compare them with the code produced from the nearest in-range parameter (Appendix G).

The answer differs by direction (Figure 3). Below the hull, the oracle-input condition closes $61.2\% \pm 7.2$ of the 111.14 ± 2.18 return gap between the nearest-boundary-input condition and the expert. Its contribution is $+68.08 \pm 8.67$, leaving a 43.06 ± 7.70 gap to the expert. Above the hull, it closes -237.3% of a 96.21 ± 3.18 gap: its contribution is -228.24 ± 64.18, while the remaining expert gap is 324.46 ± 64.58. Thus the oracle-input condition is worse for return than the boundary-input condition on that side. Although the reward $-|v - \theta|$ is symmetric in speed error, the feasible behavior is not: standing is close to a slow target, whereas top speed is limited. For targets from 3.6 to 4.0 m/s, the oracle-input policy settles at 0.803 m/s, compared with 2.988 m/s under the boundary-input condition. The SAC expert earns -61.97 on that band, so this failure is not an unavoidable property of the environment.

Widening the training range asks whether the saturation level moves. The treated variant, with three seeds like its control, adds ten exactly relabeled tasks on $[3.05, 3.50]$ to the original twenty (Lee & Chung, 2021; Kim et al., 2025). The far band $[3.6, 4.0]$, never used for relabeling, is the valid OOD test. There, achieved speed rises from 2.9809 ± 0.0063 to 3.3268 ± 0.0147, and the rollout-state probe rises from 2.9601 ± 0.0066 to 3.2997 ± 0.1119. Both move toward the new edge at 3.5 without reaching it, and the layer-0 control also moves. The far-band slope remains only 0.14 to 0.25 of the same variant's in-range slope. This one-family result shows that added coverage moves the saturation level; it does not establish a universal coverage boundary.

8 Why a convex readout cannot leave the hull

The next results describe two idealized readout classes; they do not identify the mechanism inside the trained agents. Write $\hat{\theta}(x)$ for a latent task estimate, $\hat{\theta}_{\text{att}}(x)$ for its assumed convex component, and $J_\theta(\pi)$ for return. Appendix A states every assumption. The elementary mechanism, adapted from the convex-aggregation limitation studied by Yan et al. (2026b), is that a convex combination of training task parameters cannot leave their hull. Attention itself averages value vectors, so applying this mechanism to task parameters is an explicit modeling assumption (Tsai et al., 2019).

Theorem 1 (Hull confinement and its return cost)**.** *Under Assumptions 1–4, suppose the convex component is $\hat{\theta}_{\mathrm{att}}(x) = \sum_i w_i(x)\theta_i$, the total residual has a finite uniform norm bound and outward component at most $\delta_{\mathrm{out}}(\theta)$, the prompt estimate is fixed during an episode, and absolute suboptimality grows at least linearly with estimate error near H. Then $\hat{\theta}_{\mathrm{att}}(x) \in H$, the error is at least $d(\theta) - \delta_{\mathrm{out}}(\theta)$ off the hull, and*

$$J^\star(\theta) - J_\theta\big(\pi_{\hat{\theta}(x_0)}\big) \;\geq\; c\,\max\big\{0,\; d(\theta) - \delta_{\mathrm{out}}(\theta)\big\}, \tag{1}$$

which is at least $c(1-\rho)\,d(\theta)$, linear in hull distance, whenever the outward slack is proportional, $\delta_{\mathrm{out}}(\theta) \leq \rho\, d(\theta)$ with $\rho \in [0, 1)$.

The bound is conditional. We do not measure an upper bound on δ_{out}, ρ, or c, and the instruments are proxies rather than observations of the theorem's latent $\hat{\theta}$. Equation (1) therefore explains a possible mechanism but does not yield a numerical lower bound for our agents.

The first bound would also hold for an uninformative prompt. The HalfCheetah-Vel audit motivates a separate sufficient-condition result for contexts that do contain the parameter. Fix a per-transition feature map ϕ into $\mathbb{R}^m$ and use its average $\psi(c)$ as an invariant summary (Zaheer et al., 2017); the theorem separately assumes that the n transitions are independent. Require $\mathbb{E}[\phi(z_t)] = A\theta + b$ on and off the hull, with A of full column rank. Fit (A, b) forward on calibration contexts from training tasks alone and invert it: $\mathcal{R}(c) = \widehat{A}^{+}(\psi(c) - \hat{b})$.

Theorem 2 (Affine readout, error uniform in hull distance up to a calibration term)**.** *Under Assumptions 5–8, write $\kappa = \|A^{+}\|$ and $R_H = \max_{u\in H}\|u\|$. Let $\varepsilon_n(\eta)$ and $\varepsilon_B(\eta')$ be the deviations in Appendix A, and suppose $\kappa\varepsilon_B(\eta') \leq 1/2$. With probability at least $1-\eta'$ over the calibration contexts, for every $\theta \in \Theta_{\mathrm{ev}}$ an independent $c \sim D_\theta$ satisfies, with probability at least $1-\eta$,*

$$\big\|\mathcal{R}(c) - \theta\big\| \;\leq\; 2\kappa\,\varepsilon_n(\eta) \;+\; 2\kappa\,\varepsilon_B(\eta')\big(1 + R_H + d(\theta)\big). \tag{2}$$

Independently of calibration, the population readout $\mathcal{R}^\star(c) = A^{+}(\psi(c) - b)$ satisfies $\|\mathcal{R}^\star(c) - \theta\| \leq \kappa\varepsilon_n(\eta)$ with probability at least $1-\eta$.

The population construction is not a convex average of training tasks, so its displayed error bound has no hull-distance term. The fitted version does have such a term. This theorem analyzes a forward fit of the statistic on the parameter, followed by inversion, for independent transitions. Our statsC variants instead learn $\psi \mapsto z$ jointly with the policy, and their five-step contiguous trajectory prompts are not independent and have not been shown to satisfy the theorem's noise or affine-mean assumptions. Ant-Dir's circular statistic and Ant-Goal's reward-weighted statistic also fall outside the stated affine form. The theorem is therefore a sufficient-condition construction for information in a context, not a guarantee for statsC or a direct test of the intervention.

9 Discussion and Limitations

Building on standard train/test task-distribution descriptions (Kirk et al., 2023; Beck et al., 2025), we recommend also reporting the parameter region covered in training and each evaluation task's position relative to it. Return alone cannot distinguish an incorrect task estimate from a policy that cannot use a correct one. Our strongest latent-estimate evidence is HalfCheetah-Vel; only two other families have any passing decoder, and Ant-Goal's useful probe is contaminated. The other eleven families are instrument limited; among them, Humanoid-Dir is excluded and the six MetaWorld families support only behavioral conclusions. The statistics are supplied by an analyst, the architecture variants change training as well as conditioning, and their OOD gains often cost in-range accuracy. Decoder attainment is measured on training tasks only, but held-out-band diagnostics were used to choose reported instruments and to screen statistics and families for the intervention, with part of the latter screen corrected post hoc. The estimates should therefore be treated as exploratory rather than as confirmatory results from a fully untouched test set.

A confirmatory follow-up should choose decoder layers, statistics, and family exclusions without consulting the held-out bands, then reserve a second set of outside-hull tasks for final evaluation. It should also use online probes evaluated on agent-generated rollouts rather than offline expert segments and report uncertainty over sampled tasks, not only model seeds. Testing disconnected training regions and varied policy capacities would help separate hull geometry from data coverage.

AI USE STATEMENT

We used generative AI tools to search for literature, edit language, and audit and revise mathematical statements and proofs. The tools also helped identify places where empirical claims needed narrower wording. The authors checked all cited sources, proof steps, numerical statements, and AI-assisted edits against the original papers, experiment records, and code. The authors take responsibility for the final content, including all text, claims, proofs, and artifacts produced with AI assistance.

ETHICS STATEMENT

This work trains sequence models on offline data that we collect in simulated locomotion and manipulation environments. It involves no human subjects and no personally identifiable data, and the datasets are generated entirely by scripted and learned controllers inside those simulators. The finding we report is a limitation of existing in-context agents rather than a new capability, so we are not aware of harmful applications that it enables beyond those already possible with the algorithms we evaluate, and we declare no conflicts of interest.

REPRODUCIBILITY STATEMENT

Appendices B to E fix every task family and its parameter, every split and band boundary, how each offline dataset was collected and what it contains, the hyperparameters of every arm, the parity checks behind the routing intervention, the fitting protocol of each instrument and each hand-designed statistic. Appendix H says what is released and how each printed number is traced to the run that produced it.

REFERENCES

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, volume 24, pp. 2312–2320. Curran Associates, Inc., 2011. URL https://proceedings.neurips.cc/paper_files/paper/2011/file/e1d5be1c7f2f456670de3d53c7b54f4a-Paper.pdf.

André Barreto, Will Dabney, Rémi Munos, Jonathan J. Hunt, Tom Schaul, Hado van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 30, pp. 4055–4065. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/350db081a661525235354dd3e19b8c05-Paper.pdf.

Jacob Beck, Risto Vuorio, Evan Zheran Liu, Zheng Xiong, Luisa Zintgraf, Chelsea Finn, and Shimon Whiteson. A tutorial on meta-reinforcement learning. *Foundations and Trends in Machine Learning*, 18(2–3):224–384, 2025. doi: 10.1561/2200000080. URL https://doi.org/10.1561/2200000080.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, 2004. doi: 10.1017/CBO9780511804441. URL https://doi.org/10.1017/CBO9780511804441.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Michael Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. In *Advances in Neural Information Processing Systems*, volume 34, pp. 15084–15097. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/7f489f642a0ddb10272b5c31057f0663-Paper.pdf.

Ron Dorfman, Idan Shenfeld, and Aviv Tamar. Offline meta reinforcement learning – identifiability challenges and effective data collection strategies. In *Advances in Neural Information Processing Systems*, volume 34, pp. 4607–4618. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/248024541dbda1d3fd75fe49d1a4df4d-Paper.pdf.

Finale Doshi-Velez and George Konidaris. Hidden parameter Markov decision processes: A semi-parametric regression approach for discovering latent task parametrizations. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1432–1440, 2016. URL https://www.ijcai.org/Proceedings/16/Papers/206.pdf.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4RL: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020. doi: 10.48550/arXiv.2004.07219. URL https://arxiv.org/abs/2004.07219.

Yunkai Gao, Rui Zhang, Jiaming Guo, Fan Wu, Qi Yi, Shaohui Peng, Siming Lan, Ruizhi Chen, Zidong Du, Xing Hu, Qi Guo, Ling Li, and Yunji Chen. Context shift reduction for offline meta-reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 80024–80043. Curran Associates, Inc., 2023. doi: 10.52202/075280-3505. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/fd489a44f3bcb9f122e4931ef21d0c43-Paper-Conference.pdf.

Chase Goddard, Lindsay M. Smith, Vudtiwat Ngampruetikorn, and David J. Schwab. When can in-context learning generalize out of task distribution? In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 19585–19599. PMLR, 2025. URL https://proceedings.mlr.press/v267/goddard25a.html.

Wes Gurnee and Max Tegmark. Language models represent space and time. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/0a6059857ae5c82ea9726ee9282a7145-Abstract-Conference.html.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1861–1870. PMLR, 2018. URL https://proceedings.mlr.press/v80/haarnoja18b.html.

Roee Hendel, Mor Geva, and Amir Globerson. In-context learning creates task vectors. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 9318–9333, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.624. URL https://aclanthology.org/2023.findings-emnlp.624/.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2733–2743, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1275. URL https://aclanthology.org/D19-1275/.

Michael Janner, Qiyang Li, and Sergey Levine. Offline reinforcement learning as one big sequence modeling problem. In *Advances in Neural Information Processing Systems*, volume 34, pp. 1273–1286. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/099fe6b0b444c23836c4a5d07346082b-Paper.pdf.

Jeongmo Kim, Yisak Park, Minung Kim, and Seungyul Han. Task-aware virtual training: Enhancing generalization in meta-reinforcement learning for out-of-distribution tasks. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 30720–30748. PMLR, 2025. URL https://proceedings.mlr.press/v267/kim25ag.html.

Robert Kirk, Amy Zhang, Edward Grefenstette, and Tim Rocktäschel. A survey of zero-shot generalisation in deep reinforcement learning. *Journal of Artificial Intelligence Research*, 76: 201–264, 2023. doi: 10.1613/jair.1.14174. URL https://jair.org/index.php/jair/article/view/14174.

Michael Laskin, Luyu Wang, Junhyuk Oh, Emilio Parisotto, Stephen Spencer, Richie Steigerwald, DJ Strouse, Steven Hansen, Angelos Filos, Ethan Brooks, Maxime Gazeau, Himanshu Sahni, Satinder Singh, and Volodymyr Mnih. In-context reinforcement learning with algorithm distillation. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=hy0a5MMPUv.

Jonathan N. Lee, Annie Xie, Aldo Pacchiano, Yash Chandak, Chelsea Finn, Ofir Nachum, and Emma Brunskill. Supervised pretraining can learn in-context reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 43057–43083. Curran Associates, Inc., 2023. doi: 10.52202/075280-1866. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/8644b61a9bc87bf7844750a015feb600-Paper-Conference.pdf.

Suyoung Lee and Sae-Young Chung. Improving generalization in meta-RL with imaginary tasks from latent dynamics mixture. In *Advances in Neural Information Processing Systems*, volume 34, pp. 27222–27235. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/e48e13207341b6bffb7fb1622282247b-Paper.pdf.

Kenneth Li, Aspen K. Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Emergent world representations: Exploring a sequence model trained on a synthetic task. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=DeG07_TcZvT.

Lanqing Li, Rui Yang, and Dijun Luo. FOCAL: Efficient fully-offline meta-reinforcement learning via distance metric learning and behavior regularization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=8cpHIfgY4Dj.

Lanqing Li, Hai Zhang, Xinyu Zhang, Shatong Zhu, Yang Yu, Junqiao Zhao, and Pheng-Ann Heng. Towards an information theoretic framework of context-based offline meta-reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 75642–75667. Curran Associates, Inc., 2024. doi: 10.52202/079017-2408. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/8a30aba6514b56d02976f49797f6338a-Paper-Conference.pdf.

Eric Mitchell, Rafael Rafailov, Xue Bin Peng, Sergey Levine, and Chelsea Finn. Offline meta-reinforcement learning with advantage weighting. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 7780–7791. PMLR, 2021. URL https://proceedings.mlr.press/v139/mitchell21a.html.

Mohammadreza Nakhaei, Aidan Scannell, Kevin Sebastian Luck, and Joni Pajarinen. Temporal consistency improves generalization in contextual offline meta reinforcement learning. *arXiv preprint arXiv:2603.02935*, 2026. doi: 10.48550/arXiv.2603.02935. URL https://arxiv.org/abs/2603.02935.

Mohammadreza Nakhaeinezhadfard, Aidan Scannell, and Joni Pajarinen. Entropy regularized task representation learning for offline meta-reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 19616–19623, 2025. doi: 10.1609/aaai.v39i18.34160. URL https://ojs.aaai.org/index.php/AAAI/article/view/34160.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1):3942–3951, 2018. doi: 10.1609/aaai.v32i1.11671. URL https://ojs.aaai.org/index.php/AAAI/article/view/11671.

Andrei Polubarov, Nikita Lyubaykin, Alexander Derevyagin, Artyom Grishin, Igor Saprygin, Aleksandr Serkov, Mark Averchenko, Daniil Tikhonov, Maksim Zhdanov, Alexander Nikulin, Ilya Zisman, Albina Klepach, Alexey Zemtsov, and Vladislav Kurenkov. Vintix II: Decision pre-trained transformer is a scalable in-context reinforcement learner. In *International Conference on Learning Representations*, volume 2026, pp. 56944–56997, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/5d83bdbcdeba05c276c8720a3f97a049-Abstract-Conference.html.

Fuyuan Qian, Menglong Zhang, Song Wang, and Quanying Liu. Behavior-invariant task representation learning with transformer-based world models for offline meta-reinforcement learning. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*. PMLR, 2026. URL https://icml.cc/virtual/2026/poster/61775.

Kate Rakelly, Aurick Zhou, Deirdre Quillen, Chelsea Finn, and Sergey Levine. Efficient off-policy meta-reinforcement learning via probabilistic context variables. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5331–5340. PMLR, 2019. URL https://proceedings.mlr.press/v97/rakelly19a.html.

Sharath Chandra Raparthy, Eric Hambro, Robert Kirk, Mikael Henaff, and Roberta Raileanu. Generalization to new sequential decision making tasks with in-context learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 42138–42158. PMLR, 2024. URL https://proceedings.mlr.press/v235/raparthy24a.html.

Allan Raventós, Mansheej Paul, Feng Chen, and Surya Ganguli. Pretraining task diversity and the emergence of non-Bayesian in-context learning for regression. In *Advances in Neural Information Processing Systems*, volume 36, pp. 14228–14246. Curran Associates, Inc., 2023. doi: 10.52202/075280-0626. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/2e10b2c2e1aa4f8083c37dfe269873f8-Abstract-Conference.html.

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

Yao-Hung Hubert Tsai, Shaojie Bai, Makoto Yamada, Louis-Philippe Morency, and Ruslan Salakhutdinov. Transformer dissection: An unified understanding for transformer's attention via the lens of kernel. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4344–4353, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1443. URL https://aclanthology.org/D19-1443/.

Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*, volume 47 of *Cambridge Series in Statistical and Probabilistic Mathematics*. Cambridge University Press, 2018. ISBN 9781108415194. doi: 10.1017/9781108231596. URL https://doi.org/10.1017/9781108231596.

Qixun Wang, Yifei Wang, Xianghua Ying, and Yisen Wang. Can in-context learning really generalize to out-of-distribution tasks? In *International Conference on Learning Representations*, volume 2025, pp. 83553–83574, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/cf7a83a5342befd11d3d65beba1be5b0-Abstract-Conference.html.

Zhi Wang, Li Zhang, Wenhao Wu, Yuanheng Zhu, Dongbin Zhao, and Chunlin Chen. Meta-DT: Offline meta-RL as conditional sequence modeling with world model disentanglement. In *Advances in Neural Information Processing Systems*, volume 37, pp. 44845–44870. Curran Associates, Inc., 2024. doi: 10.52202/079017-1425. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/4f3820576130a8f796ddbf204c841487-Abstract-Conference.html.

Wenhao Wu, Fuhong Liu, Haoru Li, Zican Hu, Daoyi Dong, Chunlin Chen, and Zhi Wang. Mixture-of-experts meets in-context reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 24751–24785. Curran Associates, Inc., 2025. doi: 10.52202/085713-0832. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/233cbc888ef01917d8d895e471e80a01-Abstract-Conference.html.

Zixuan Xie, Xinyu Liu, Claire Chen, Shuze Daniel Liu, Rohan Chandra, and Shangtong Zhang. Beyond linear attention: Softmax transformers implement in-context reinforcement learning, 2026. URL https://arxiv.org/abs/2605.07333.

Mengdi Xu, Yikang Shen, Shun Zhang, Yuchen Lu, Ding Zhao, Joshua B. Tenenbaum, and Chuang Gan. Prompting decision transformer for few-shot policy generalization. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 24631–24645. PMLR, 2022. URL https://proceedings.mlr.press/v162/xu22g.html.

Steve Yadlowsky, Lyric Doshi, and Nilesh Tripuraneni. Pretraining data mixtures enable narrow model selection capabilities in transformer models, 2023. URL https://arxiv.org/abs/2311.00871.

Hao Yan, Haolin Yang, and Yiqiao Zhong. Task vector geometry underlies dual modes of task inference in transformers, 2026a. URL https://arxiv.org/abs/2605.03780.

Jingquan Yan, Yuwei Miao, Peiran Yu, and Junzhou Huang. Breaking the correlation plateau: On the optimization and capacity limits of attention-based regressors. In *International Conference on Learning Representations*, volume 2026, pp. 125984–126018, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/cc2c03ce028a1d6cbed6f98cede7f5a9-Abstract-Conference.html.

Qingmao Yao, Zhichao Lei, Tianyuan Chen, Ziyue Yuan, Xuefan Chen, Jianxiang Liu, Faguo Wu, and Xiao Zhang. Offline RL with smooth OOD generalization in convex hull and its neighborhood. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/792a4eb351b014e1d61bcfaab0ddd555-Abstract-Conference.html.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-World: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Proceedings of the Conference on Robot Learning (CoRL)*, volume 100 of *Proceedings of Machine Learning Research*, pp. 1094–1100. PMLR, 2020. URL https://proceedings.mlr.press/v100/yu20a.html.

Haoqi Yuan and Zongqing Lu. Robust task representations for offline meta-reinforcement learning via contrastive learning. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 25747–25759. PMLR, 2022. URL https://proceedings.mlr.press/v162/yuan22a.html.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabás Póczos, Ruslan Salakhutdinov, and Alexander J. Smola. Deep sets. In *Advances in Neural Information Processing Systems*, volume 30, pp. 3391–3401. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/f22e4747da1aa27e363d86d40ff442fe-Abstract.html.

Renzhe Zhou, Chen-Xiao Gao, Zongzhang Zhang, and Yang Yu. Generalizable task representation learning for offline meta-reinforcement learning with data limitations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(15):17132–17140, 2024. doi: 10.1609/aaai.v38i15.29658. URL https://ojs.aaai.org/index.php/AAAI/article/view/29658.

Luisa Zintgraf, Kyriacos Shiarlis, Maximilian Igl, Sebastian Schulze, Yarin Gal, Katja Hofmann, and Shimon Whiteson. VariBAD: A very good method for Bayes-adaptive deep RL via meta-learning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=Hkl9JlBYvr.

A Proofs

This section states the assumptions of Theorems 1 and 2 in full and proves them. It also explains why the architecture intervention in Section 6 is not a direct test of either theorem. The theorem numbering restarts here so that the restatements carry the numbers used in the main text.

What the two theorems do together. Theorem 1 idealizes a latent readout as a convex average of training-task parameters plus a bounded residual. Under further policy-conditioning and absolute-suboptimality assumptions, it gives error and return-loss floors. Theorem 2 analyzes a different, explicitly constructed readout: a forward affine fit of a hand-designed average over independent transitions. Its population error bound is independent of hull distance, while its finite-calibration bound is not. These are conditional comparisons of readout classes. They do not show that a trained transformer's latent state has the assumed form, that the experimental contexts meet the concentration assumptions. Nor do they show that the separately trained statsC architecture implements the analyzed affine inverse.

Notation shared by the two proofs. Task parameters live in $\Theta \subseteq \mathbb{R}^d$ with the Euclidean norm $\|\cdot\|$ and inner product $\langle\cdot,\cdot\rangle$. There are N training tasks $\theta_1, \dots, \theta_N$, their convex hull is $H := \mathrm{conv}\{\theta_1, \dots, \theta_N\}$, and the hull distance of a parameter is

$$d(\theta) := \min_{u \in H} \|\theta - u\|.$$

The evaluation set is Θ_{ev}; it contains H and the held-out tasks, and we assume it is bounded. The letter n always counts transitions in one context and never counts tasks. Theorem 1 writes its slack as δ and its proportional slack as ρ; Theorem 2 writes its failure probabilities as η and η' and its conditioning constant as κ.

A.1 Theorem 1: hull confinement

Setup. For a policy π let $J_\theta(\pi)$ be its expected return in the MDP with parameter θ over the evaluation horizon, and let $J^\star(\theta) := \sup_\pi J_\theta(\pi)$, assumed finite. A *prompt* x_0 is the block of demonstration transitions the agent is given before the episode starts. A *context* x is what the agent conditions on at a given step, that is, the prompt together with the part of the current episode already seen. We write $\mathcal{X}_0(\theta)$ for the prompts and $\mathcal{X}(\theta) \supseteq \mathcal{X}_0(\theta)$ for all contexts that can arise when the agent is deployed in task θ, and $\hat\theta(x) \in \mathbb{R}^d$ for the agent's implicit task estimate on context x, the quantity our three instruments decode.

The four assumptions are separate on purpose, so that Remark 1 can say which one each part of the gap between theory and measurement sits in.

Assumption 1 (Convex readout of the cross-episode path)**.** The cross-episode attention path of the agent produces, on every context x, an estimate that is a convex combination of the training task parameters:

$$\hat\theta_{\mathrm{att}}(x) = \sum_{i=1}^{N} w_i(x)\,\theta_i, \qquad w_i(x) \geq 0, \qquad \sum_{i=1}^{N} w_i(x) = 1,$$

where the support points $\theta_1, \dots, \theta_N$ are the training tasks and do not depend on x.

The proof uses only that the weights are nonnegative and sum to one. Softmax is one way to produce such weights and nothing below depends on it, so the result covers any convex aggregator. What Tsai et al. (2019) establish, though, is that attention is a kernel smoother over the *value vectors* attached to the attended items, not over task parameters. Turning that into the display above needs one more step, that the value attached to a training task and the readout that follows the attention compose to the identity on training tasks. That step is ours and it is an idealization, not a citation; Remark 1 says exactly when it is exact.

Assumption 2 (Bounded slack from the non-attention path)**.** There are functions $\delta, \delta_{\mathrm{out}} : \Theta_{\mathrm{ev}} \to [0, \infty)$ with $\delta_{\mathrm{out}} \leq \delta$ such that, for every $\theta \in \Theta_{\mathrm{ev}}$ and every $x \in \mathcal{X}(\theta)$,

$$\big\|\hat\theta(x) - \hat\theta_{\mathrm{att}}(x)\big\| \leq \delta(\theta), \qquad \text{and, if } \theta \notin H, \qquad \big\langle g(\theta),\, \hat\theta(x) - \hat\theta_{\mathrm{att}}(x)\big\rangle \leq \delta_{\mathrm{out}}(\theta),$$

where $g(\theta)$ is the unit outward direction of Lemma 1(iii). Assume explicitly that $\bar\delta := \sup_{\theta \in \Theta_{\mathrm{ev}}} \delta(\theta) < \infty$, and let $H_{\bar\delta} := \{u \in \mathbb{R}^d : \min_{v \in H} \|u - v\| \leq \bar\delta\}$.

The second bound is the one the estimation floor uses. In one-dimensional and ray-aligned protocols, our instruments target this outward coordinate; MetaWorld instead reports a fixed band-axis proxy. It is implied by the first with $\delta_{\mathrm{out}} = \delta$, so nothing is lost by carrying both. The first bound is still needed, because it is what places the estimate in $H_{\bar\delta}$, the set on which Assumption 4 is stated.

Assumption 3 (Estimate to policy)**.** There is a map $\theta' \mapsto \pi_{\theta'}$, the conditioning the agent learned at training time, such that, for every $\theta \in \Theta_{\mathrm{ev}}$ and $x_0 \in \mathcal{X}_0(\theta)$, the agent executes $\pi_{\hat{\theta}(x_0)}$. The estimate is therefore read off the prompt and is held fixed for the episode. We assume every return displayed below is finite; for each θ, the map $\theta' \mapsto J_\theta(\pi_{\theta'})$ is measurable, and $J_\theta(\pi_{\hat{\theta}(x_0)})$ is integrable whenever x_0 is random.

Assumption 4 (Absolute suboptimality sharpness near the hull)**.** There is a constant $c > 0$ such that

$$J^\star(\theta) - J_\theta(\pi_{\theta'}) \;\geq\; c\,\|\theta - \theta'\| \qquad \text{for every } \theta \in \Theta_{\mathrm{ev}} \text{ and every } \theta' \in H_{\bar{\delta}}.$$

The constant belongs to the pair (family, conditioning map of Assumption 3), not to the family alone.

Sharpness is asked for only on $H_{\bar{\delta}}$. That restriction matters: a fixed linear lower bound need not hold over an unbounded domain for the estimate. Confinement is therefore what allows sharpness to be required only where the estimate can actually be located.

Lemma 1 (Confinement and the distance floor)**.** *Let $\theta \in \Theta$ and let w be any vector with $w_i \geq 0$ and $\sum_i w_i = 1$. Then:*

(i) $\sum_{i=1}^N w_i\theta_i \in H$*;*

(ii) $\|\theta - u\| \geq d(\theta)$ *for every $u \in H$, and the minimum defining $d(\theta)$ is attained at a unique point $P_H(\theta) \in H$;*

(iii) if $\theta \notin H$ and $g(\theta) := \big(\theta - P_H(\theta)\big)/d(\theta)$, then $\|g(\theta)\| = 1$ and $\langle g(\theta),\, \theta - u\rangle \geq d(\theta)$ for every $u \in H$.

Proof. (i) The convex hull of a finite set is by definition the set of all convex combinations of its points.

(ii) H is the image of the probability simplex in $\mathbb{R}^N$, which is compact, under the continuous map $w \mapsto \sum_i w_i\theta_i$, so H is nonempty and compact. The function $u \mapsto \|\theta - u\|$ is continuous, so it attains its minimum on H, and every $u \in H$ satisfies $\|\theta - u\| \geq d(\theta)$. The minimizer is unique because H is convex and $u \mapsto \|\theta - u\|^2$ is strictly convex; this is the projection theorem for closed convex sets (Boyd & Vandenberghe, 2004, §8.1).

(iii) Convexity of H gives $\langle \theta - P_H(\theta),\, u - P_H(\theta)\rangle \leq 0$ for every $u \in H$: writing $p = P_H(\theta)$, the point $p + t(u - p)$ lies in H for every $t \in [0, 1]$, so $t \mapsto \|\theta - p - t(u - p)\|^2$ is minimized at $t = 0$, and its derivative there, $-2\langle \theta - p,\, u - p\rangle$, is therefore nonnegative. Since $\theta \notin H$ we have $d(\theta) > 0$, so $g(\theta)$ is well defined and has unit norm, and dividing by $d(\theta)$ gives $\langle g(\theta), u - P_H(\theta)\rangle \leq 0$. Hence, for every $u \in H$, and writing g for $g(\theta)$,

$$\langle g,\, \theta - u\rangle = \langle g,\, \theta - P_H(\theta)\rangle + \langle g,\, P_H(\theta) - u\rangle = d(\theta) - \langle g,\, u - P_H(\theta)\rangle \;\geq\; d(\theta). \qquad \square$$

Part (iii) is the form the theorem uses. It says more than (ii): the error is not merely large in norm, it has a component of size at least $d(\theta)$ along one fixed direction, the outward normal. It matches the reported axis only in one-dimensional or ray-aligned protocols; for MetaWorld the reported band axis need not equal $g(\theta)$.

Lemma 2 (Estimation error to return suboptimality)**.** *Assume 3 and 4. Let $\theta \in \Theta_{\mathrm{ev}}$ and let the agent run an episode in task θ whose prompt gives an estimate $\hat{\theta} \in H_{\bar{\delta}}$. Then*

$$J^\star(\theta) - J_\theta(\pi_{\hat{\theta}}) \;\geq\; c\,\|\theta - \hat{\theta}\| \;\geq\; 0,$$

and if the prompt, and hence $\hat{\theta}$, is random, then $J^\star(\theta) - \mathbb{E}\big[J_\theta(\pi_{\hat{\theta}})\big] \geq c\,\mathbb{E}\big[\|\theta - \hat{\theta}\|\big]$.

Proof. By Assumption 3 the policy executed on the episode is $\pi_{\hat{\theta}}$, and $\hat{\theta}$ is fixed before the episode starts, so the expected return of the episode given the prompt is $J_\theta(\pi_{\hat{\theta}})$. Since $\hat{\theta} \in H_{\bar{\delta}}$, Assumption 4 applies with $\theta' = \hat{\theta}$ and gives the first inequality. The second holds because $J^\star(\theta)$ is a supremum over all policies. For the expectation form, take expectations over the prompt on both sides of thsi bound; expectation preserves the inequality and $J^\star(\theta)$ is a constant. $\square$

Assumption 4 is a lower bound on absolute suboptimality, not the more familiar Lipschitz upper bound. The direction matters. An upper bound of the form $J^\star(\theta) - J_\theta(\pi_{\theta'}) \le L_J\|\theta - \theta'\|$ says a good estimate is enough for good return, and it cannot by itself produce a loss lower bound. Theorem 1 lower-bounds a loss, so it needs the lower bound. When both hold, estimate error is the right currency for return, with the two constants c and L_J setting the exchange rate.

Theorem 1 (Hull confinement and its return cost)**.** *Assume 1, 2, 3 and 4. Let $\theta \in \Theta_{\mathrm{ev}}$. Then:*

(i) (confinement) *for every context $x \in \mathcal{X}(\theta)$, $\hat\theta_{\mathrm{att}}(x) \in H$ and $\hat\theta(x) \in H_{\bar\delta}$;*

(ii) (estimation floor) *if $\theta \notin H$ and $g = g(\theta)$ is the unit outward direction of Lemma 1(iii), then for every $x \in \mathcal{X}(\theta)$,*

$$\big\|\theta - \hat\theta(x)\big\| \;\ge\; \big\langle g,\, \theta - \hat\theta(x)\big\rangle \;\ge\; d(\theta) - \delta_{\mathrm{out}}(\theta);$$

(iii) (return floor) *with $x_0 \in \mathcal{X}_0(\theta)$ the prompt of an episode in task θ,*

$$J^\star(\theta) - J_\theta\big(\pi_{\hat\theta(x_0)}\big) \;\ge\; c\,\max\big\{0,\; d(\theta) - \delta_{\mathrm{out}}(\theta)\big\},$$

and the same bound holds for $J^\star(\theta) - \mathbb{E}[J_\theta(\pi_{\hat\theta(x_0)})]$ when the prompt is random.

If $\delta_{\mathrm{out}}(\theta) \le \delta_\star$ for all θ, the floor is at least $c\,\max\{0, d(\theta) - \delta_\star\}$. If it is proportional, $\delta_{\mathrm{out}}(\theta) \le \rho\, d(\theta)$ with $\rho \in [0,1)$, the floor is at least $c\,(1-\rho)\, d(\theta)$, which grows linearly with hull distance.

Proof. (i) The first claim is Lemma 1(i) applied to the weights $w(x)$ of Assumption 1, which are nonnegative and sum to one. For the second, Assumption 2 gives $\|\hat\theta(x) - \hat\theta_{\mathrm{att}}(x)\| \le \delta(\theta) \le \bar\delta$, and $\hat\theta_{\mathrm{att}}(x) \in H$, so $\hat\theta(x)$ is within $\bar\delta$ of H.

(ii) Write $e := \hat\theta(x) - \hat\theta_{\mathrm{att}}(x)$, so $\langle g, e\rangle \le \delta_{\mathrm{out}}(\theta)$ by Assumption 2. Apply Lemma 1(iii) with $u = \hat\theta_{\mathrm{att}}(x)$, which lies in H by (i), to get $\langle g, \theta - \hat\theta_{\mathrm{att}}(x)\rangle \ge d(\theta)$. Then

$$\big\langle g,\, \theta - \hat\theta(x)\big\rangle = \big\langle g,\, \theta - \hat\theta_{\mathrm{att}}(x)\big\rangle - \langle g, e\rangle \;\ge\; d(\theta) - \delta_{\mathrm{out}}(\theta).$$

The first inequality of (ii) is Cauchy-Schwarz in the form $\|v\| \ge \langle g, v\rangle$ for any v and any unit g.

(iii) By (i) we have $\hat\theta(x_0) \in H_{\bar\delta}$, so Lemma 2 applies and gives $J^\star(\theta) - J_\theta(\pi_{\hat\theta(x_0)}) \ge c\|\theta - \hat\theta(x_0)\| \ge 0$. If $\theta \in H$ then $d(\theta) = 0$, so the right side of (iii) is zero and the claim is that nonnegativity. If $\theta \notin H$ then (ii) gives $\|\theta - \hat\theta(x_0)\| \ge d(\theta) - \delta_{\mathrm{out}}(\theta)$, and combining the two displays gives the bound; together with nonnegativity this is the stated form with the maximum. The expectation version follows from the expectation form of Lemma 2 and $\mathbb{E}\|\theta - \hat\theta(x_0)\| \ge \max\{0, d(\theta) - \delta_{\mathrm{out}}(\theta)\}$, which holds pointwise. The two special cases follow by substituting the stated bounds on δ_{out}. □

For $\theta \in H$ the bound is zero and the theorem says nothing. That is the correct behavior. Inside the hull a convex readout can be exact, and the theorem imposes no positive error floor there. The content of the theorem is entirely about $d(\theta) > 0$, and it is informative only while the slack stays below the hull distance.

Corollary 1 (Boundary-anchored bound along an outward ray)**.** *Let u_0 be a boundary point of H and let g be a unit outward normal there, meaning $\langle g, u - u_0\rangle \le 0$ for all $u \in H$; such a g exists by the supporting hyperplane theorem (Boyd & Vandenberghe, 2004, §2.5.2). For $s > 0$ put $\theta(s) := u_0 + s\,g$ and assume $\theta(s) \in \Theta_{\mathrm{ev}}$. Then $d(\theta(s)) = s$ and $g(\theta(s)) = g$, and under Assumptions 1 and 2, for every context $x \in \mathcal{X}(\theta(s))$,*

$$\big\langle g,\, \hat\theta(x)\big\rangle \;\le\; \langle g, u_0\rangle + \delta_{\mathrm{out}}(\theta(s)).$$

Hence the outward coordinate cannot exceed the boundary value by more than the outward slack, and

$$\frac{\langle g, \hat\theta(x)\rangle - \langle g, u_0\rangle}{s} \;\le\; \frac{\delta_{\mathrm{out}}(\theta(s))}{s}.$$

With uniform slack, this boundary-anchored normalized excess is at most $\delta_\star/s$, which decreases as $1/s$ over the part of the ray contained in Θ_{ev}. With proportional slack $\delta_{\mathrm{out}}(\theta(s)) \le \rho s$, it is at most ρ. These statements do not bound a free-intercept regression slope across evaluation tasks.

Proof. For any $u \in H$ we have $\langle \theta(s) - u_0, u - u_0 \rangle = s \langle g, u - u_0 \rangle \le 0$, so $\|\theta(s) - u\|^2 = \|\theta(s) - u_0\|^2 - 2\langle \theta(s) - u_0, u - u_0 \rangle + \|u - u_0\|^2 \ge \|\theta(s) - u_0\|^2$, and $u_0 \in H$, so u_0 is the Euclidean projection of $\theta(s)$ onto H, which is unique because H is closed and convex (Boyd & Vandenberghe, 2004, §8.1); hence $P_H(\theta(s)) = u_0$, $d(\theta(s)) = \|sg\| = s$ and $g(\theta(s)) = (\theta(s) - u_0)/s = g$. By Assumption 1 and $\langle g, \theta_i - u_0 \rangle \le 0$ for each training task,

$$\langle g, \hat{\theta}_{\mathrm{att}}(x) \rangle = \sum_{i=1}^{N} w_i(x) \langle g, \theta_i \rangle \le \sum_{i=1}^{N} w_i(x) \langle g, u_0 \rangle = \langle g, u_0 \rangle,$$

using $w_i(x) \ge 0$ and $\sum_i w_i(x) = 1$. Adding the outward slack of Assumption 2 gives $\langle g, \hat{\theta}(x) \rangle \le \langle g, \hat{\theta}_{\mathrm{att}}(x) \rangle + \delta_{\mathrm{out}}(\theta(s)) \le \langle g, u_0 \rangle + \delta_{\mathrm{out}}(\theta(s))$. The normalized-excess statements follow by subtracting $\langle g, u_0 \rangle$ and dividing by $s > 0$. □

Remark 1 (What the idealization assumes away). Assumption 1 treats the cross-episode path as a weighted average of stored task parameters. Real attention averages value vectors, not task labels, and the map from a value vector to a task parameter is learned rather than given. The assumption is exact in one case: if the value attached to a training task is an affine function $V\theta_i + v_0$ of its parameter, and the readout that follows the attention is affine and inverts that map on training tasks, then the composition returns $\sum_i w_i \theta_i$ exactly, because an affine map sends a convex combination to the convex combination of the images. Away from that case it is an approximation. The assumption also puts the training tasks in the support, which is a claim about where the readout gets its values, not a claim about the context. The HalfCheetah-Vel model-free audit shows that its context contains target-speed information. It does not verify the assumed convex decomposition or establish the same fact for every family.

Assumption 2 is where everything non-convex is parked. A real block has residual connections, layer normalization, and MLP layers, and none of them is a convex aggregator. They can move the estimate off the hull. The slack is a budget we assume, not a quantity we derive, and the theorem is empty once $\delta_{\mathrm{out}}(\theta)$ reaches $d(\theta)$. Corollary 1 bounds a boundary-anchored ratio for the latent estimate. It does not bound the free-intercept band regressions in Table 10; doing so would require additional conditions such as boundary attainment and monotonicity. Nor do our instruments reveal $\hat{\theta}$ without error: the probe, action inversion, and behavior readout are distinct proxies with explicit quality checks. We therefore do not substitute their measured errors into Lemma 2. In addition, the theorem would need empirical upper bounds on δ_{out} and a value of c to produce a numerical return floor, and we do not estimate either quantity.

The theorem gives no lower bound on the estimate's outward coordinate, so it does not claim that the estimate reaches the boundary. It only upper-bounds how far that coordinate can pass the boundary. Some of our instruments show a slightly negative beyond-hull slope, meaning the estimate drifts back inward, which the theorem allows.

Assumption 3 treats the agent as a policy conditioned on one estimate, read off the prompt and held fixed for the episode. A real agent re-reads its context at every step, and nothing forces its behavior to equal that of any conditioned policy. Without this assumption $J_\theta(\pi_{\hat{\theta}})$ is not even well defined, since the estimate would depend on the trajectory the executed policy generates. Our action-inversion instrument is the empirical stand-in: it asks which conditioned expert the emitted actions look like, and reports how well any single one fits.

Assumption 4 states that absolute suboptimality grows at a linear rate with parameter mismatch. Take a velocity family with per-step reward $-|v_t - \theta|$, where every target in the range can be tracked exactly from the first step and there is no control cost. Then $J^\star(\theta) = 0$ and $J_\theta(\pi_{\theta'}) = -T|\theta - \theta'|$ over a horizon of T steps, so the assumption holds with $c = T$, or with $c = 1$ if returns are reported per step. Once tracking takes time or costs control effort, the difference between the control cost of the optimal policy and that of the mismatched policy can take either sign, so c has to be measured rather than derived directly. In a family where far apart parameters call for nearly the same behavior, c is small and the bound is weak. The constant also depends on the learned conditioning map, not on the family alone, which is why we do not call it a family constant.

Finally, a confined estimate and Assumption 4 make the return-floor calculation short. The demanding steps are verifying that the task-parameter readout is convex and establishing c. This paper does neither directly, so the result is a conditional explanation, not an empirical certificate of the reported return gaps.

Remark 2 (Relation to prior work). The skeleton of the argument, that a convex aggregator cannot leave the convex hull of what it aggregates and that this caps what the aggregator can express, is due to Yan et al. (2026b). They work in supervised set-to-scalar regression, where each sample is a set of element embeddings, any convex aggregator produces a sample embedding inside the convex hull of those embeddings, and a linear head reads a scalar off it. Their capacity result bounds the gain in batch Pearson correlation of any convex aggregator over plain mean pooling by a ratio whose numerator is the root mean square radius of the within-sample embedding hull, under a condition that this radius is smaller than the spread of the mean-pooling predictions. Their fix rescales the residual around the mean embedding by a learned nonnegative factor. We take that skeleton as given and claim no part of it. What is new here is the object and the conclusion. The hull is the hull of training task parameters, formed across episodes in a sequential decision problem, rather than the hull of token embeddings within one sample. Our result lower-bounds estimation error by $d(\theta) - \delta_{\mathrm{out}}(\theta)$ and return suboptimality by its positive part times c. Proportional outward slack yields a lower bound linear in hull distance. Our instruments do not estimate these theorem quantities without error. Their setting has no tasks, no episodes and no held-out parameters, so neither the object nor the conclusion is available there.

A.2 Theorem 2: an affine readout extrapolates

Setup. Fix one family with task parameter $\theta \in \Theta \subset \mathbb{R}^d$. A context is a sequence of transitions $c = (z_1, \dots, z_n)$ with $z_t \in \mathcal{Z}$, collected in task θ by a fixed prompt-collection policy. Write D_θ for the law of c. Let $\phi : \mathcal{Z} \to \mathbb{R}^m$ be a fixed per-transition feature map, designed by hand for the family, and let

$$\psi(c) \;=\; \frac{1}{n}\sum_{t=1}^{n} \phi(z_t) \;\in\; \mathbb{R}^m \tag{3}$$

be the context statistic. Write $R_H = \max_{u \in H}\|u\|$, which is attained at a training task. Vector norms are Euclidean and matrix norms are the operator norm. For any matrix M, let M^+ denote its Moore–Penrose pseudoinverse. When M has full column rank, $M^+ = (M^\top M)^{-1}M^\top$; $s_{\min}(M)$ is its smallest singular value. All contexts below, for calibration and for evaluation, have the same length n and use the same ϕ.

Assumption 5 (Affine statistic on and off the hull)**.** There exist $A \in \mathbb{R}^{m\times d}$ and $b \in \mathbb{R}^m$, both independent of θ and of t, such that

$$\mathbb{E}_{c\sim D_\theta}[\phi(z_t)] \;=\; A\theta + b \qquad \text{for every } \theta \in \Theta_{\mathrm{ev}} \text{ and every } t \le n, \tag{4}$$

and A has full column rank d, so $m \ge d$. Write $\kappa = \|A^+\| = 1/s_{\min}(A)$.

The identity is stated per transition rather than only for the average. The centered features $\xi_t = \phi(z_t) - A\theta - b$ must be mean zero one at a time for Assumption 6. The independence assumption below is satisfied, for example, when a prompt takes one transition from each of n independently collected episodes. It is not satisfied by the contiguous five-step prompts in our experiments. Averaging (4) gives $\mathbb{E}[\psi(c)] = A\theta + b$.

Assumption 6 (Sub-Gaussian context noise)**.** With $\xi_t = \phi(z_t) - A\theta - b$, there is a single $\sigma > 0$ such that for every $\theta \in \Theta_{\mathrm{ev}}$, every t, every unit vector $u \in \mathbb{R}^m$ and every $\lambda \in \mathbb{R}$, $\mathbb{E}\exp(\lambda\langle u, \xi_t\rangle) \le \exp(\lambda^2\sigma^2/2)$. The same σ is required on and off the hull.

Assumption 7 (Independence inside a context)**.** For every $\theta \in \Theta_{\mathrm{ev}}$, the transitions $z_1, \dots, z_n$ are independent under D_θ.

Assumption 8 (Calibration design, training tasks only)**.** The readout is fit on training tasks. For each training task $j \in \{1, \dots, N\}$ we hold K contexts $c_{j,k} \sim D_{\theta_j}$, independent across the pairs (j,k), and independent of the evaluation context. Write $u_j = (\theta_j, 1) \in \mathbb{R}^{d+1}$ and $G = \frac{1}{N}\sum_{j=1}^{N} u_j u_j^\top$. We assume $\gamma := \lambda_{\min}(G) > 0$, that is, the training tasks affinely span $\mathbb{R}^d$.

Assumption 8 is a statement about the training tasks alone. An affine map is pinned down by its values on any affinely spanning set, and the training tasks are themselves points of H. No off-hull task is needed to fit the readout, which matches the protocol used throughout the paper, where every decoder is fit on training tasks only. Independence of the calibration contexts from the evaluation context is used in the proof, where an evaluation event is taken on top of a calibration event.

Write $B = [A, b] \in \mathbb{R}^{m\times(d+1)}$, so that (4) reads $\mathbb{E}[\psi(c)] = Bu$ with $u = (\theta, 1)$. The readout is fit forward, statistic on parameter, and then inverted:

$$\widehat{B} \;=\; \operatorname*{argmin}_{B'\in\mathbb{R}^{m\times(d+1)}} \sum_{j=1}^{N}\sum_{k=1}^{K} \big\|\psi(c_{j,k}) - B' u_j\big\|^2, \qquad \mathcal{R}(c) \;=\; \widehat{A}^{+}\big(\psi(c) - \hat{b}\big), \tag{5}$$

where $\widehat{A}$ is the first d columns of $\widehat{B}$ and $\hat{b}$ is the last column. The map $\mathcal{R}$ is this section's version of $\hat{\theta}$. It has a separate symbol because it is a mathematical construction, not the learned readout in Assumption 1 or the statsC module used in the experiments. Remark 4 explains why the fit runs in this direction.

A.2.1 Concentration of the statistic

Lemma 3 (Nets). *Let $\mathcal{N}$ be a $1/4$-net of the unit sphere of $\mathbb{R}^m$ and $\mathcal{M}$ a $1/4$-net of the unit sphere of $\mathbb{R}^p$, with $|\mathcal{N}| \le 9^m$ and $|\mathcal{M}| \le 9^p$; such nets exist (Vershynin, 2018, Cor. 4.2.13). Then $\|x\| \le \frac{4}{3}\max_{v\in\mathcal{N}}\langle x, v\rangle$ for every $x \in \mathbb{R}^m$, and $\|Z\| \le 2\max_{v\in\mathcal{N},w\in\mathcal{M}} v^\top Z w$ for every $Z \in \mathbb{R}^{m\times p}$.*

Proof. For $x \neq 0$ put $\bar{x} = x/\|x\|$ and pick $v \in \mathcal{N}$ with $\|\bar{x} - v\| \le 1/4$. Then $\langle x, v\rangle = \|x\| + \langle x, v - \bar{x}\rangle \ge \|x\|(1 - 1/4)$; for $x = 0$ both sides vanish. For the second claim, the operator norm is attained at unit vectors x, y with $\|Z\| = y^\top Z x$. Pick $w \in \mathcal{M}$ and $v \in \mathcal{N}$ within $1/4$ of x and y. Then $y^\top Z x = v^\top Z w + y^\top Z(x - w) + (y - v)^\top Z w \le v^\top Z w + \|Z\|/2$. □

Lemma 4 (Deviation of the statistic, independent transitions). *Under Assumptions 5, 6, and 7, for any $\eta \in (0, 1)$ and any $\theta \in \Theta_{\mathrm{ev}}$, with probability at least $1 - \eta$ over $c \sim D_\theta$,*

$$\big\|\psi(c) - (A\theta + b)\big\| \;\le\; \varepsilon_n(\eta) \;:=\; 3\sigma\sqrt{\frac{m + \log(1/\eta)}{n}}. \tag{6}$$

Proof. Write $\bar{\xi} = \psi(c) - (A\theta + b) = \frac{1}{n}\sum_t \xi_t$, which is legitimate by (4). Fix a unit vector v. The variables $\langle v, \xi_t\rangle$ are independent and satisfy $\mathbb{E}\exp(\lambda\langle v, \xi_t\rangle) \le \exp(\lambda^2\sigma^2/2)$, so $\mathbb{E}\exp(\lambda\langle v, \bar{\xi}\rangle) = \prod_t \mathbb{E}\exp((\lambda/n)\langle v, \xi_t\rangle) \le \exp(\lambda^2\sigma^2/(2n))$, and Markov's inequality applied to $\exp(\lambda\langle v, \bar{\xi}\rangle)$ with $\lambda = ns/\sigma^2$ gives $\Pr(\langle v, \bar{\xi}\rangle > s) \le \exp(-ns^2/(2\sigma^2))$. By Lemma 3, $\|\bar{\xi}\| \le \frac{4}{3}\max_{v\in\mathcal{N}}\langle v, \bar{\xi}\rangle$, so a union bound over $\mathcal{N}$ gives, with probability at least $1 - \eta$, $\|\bar{\xi}\| \le \frac{4}{3}\sigma\sqrt{2\,(m\log 9 + \log(1/\eta))/n}$. Because $\log 9 > 1$, the right side is at most $\frac{4}{3}\sqrt{2\log 9}\,\sigma\sqrt{(m + \log(1/\eta))/n}$, and $\frac{4}{3}\sqrt{2\log 9} < 2.80 < 3$, which gives (6). □

A.2.2 The least-squares readout

Lemma 5 (Least-squares readout and its deviation). *Let Assumptions 5 to 8 hold. Then the minimizer in (5) is unique and equals $\widehat{B} = \big(\sum_{j,k}\psi(c_{j,k})u_j^\top\big)S^{-1}$ with $S = K\sum_j u_j u_j^\top = NKG$, and for any $\eta' \in (0, 1)$, with probability at least $1 - \eta'$ over the calibration contexts,*

$$\big\|\widehat{B} - B\big\| \;\le\; \varepsilon_B(\eta') \;:=\; 5\,\sigma\sqrt{\frac{m + d + 1 + \log(1/\eta')}{nNK\gamma}}. \tag{7}$$

If in addition $\kappa\,\varepsilon_B(\eta') \le 1/2$, then on the same event $\widehat{A}$ has full column rank and $\|\widehat{A}^{+}\| \le 2\kappa$.

Proof. Uniqueness and the closed form are the normal equations for a fixed design, invertible because $\lambda_{\min}(S) = NK\gamma > 0$ by Assumption 8. Subtracting $B = BSS^{-1}$ gives $\widehat{B} - B = \big(\sum_{j,k} e_{j,k}u_j^\top\big)S^{-1}$ with $e_{j,k} = \psi(c_{j,k}) - Bu_j$, which is mean zero by (4).

The $e_{j,k}$ are independent across (j, k) by Assumption 8, and by the first display in the proof of Lemma 4 each is directionally sub-Gaussian with proxy $\sigma_n^2 = \sigma^2/n$. Fix unit $v \in \mathbb{R}^m$ and $w \in \mathbb{R}^{d+1}$. Then $v^\top(\widehat{B} - B)w = \sum_{j,k}\langle v, e_{j,k}\rangle\,\alpha_j$ with scalar weights $\alpha_j = u_j^\top S^{-1}w$, and $\sum_{j,k}\alpha_j^2 = w^\top S^{-1}\big(K\sum_j u_j u_j^\top\big)S^{-1}w = w^\top S^{-1}w \le 1/(NK\gamma)$. Multiplying MGFs as before, the sum is sub-Gaussian with proxy $\sigma_n^2/(NK\gamma)$. Applying Lemma 3 to $\widehat{B} - B$ and

taking a union bound over the two nets gives, with probability at least $1-\eta'$, $\|\widehat{B}-B\| \le 2\sigma_n\sqrt{2\left((m+d+1)\log 9+\log(1/\eta')\right)/(NK\gamma)}$, and $2\sqrt{2\log 9} < 4.20 < 5$ gives (7). This is the standard fixed-design least-squares deviation bound. For predictable adaptive covariates and conditionally sub-Gaussian coordinate noise, the self-normalized scalar-response bound of Abbasi-Yadkori et al. (2011) can instead be applied coordinate by coordinate, followed by a union bound.

For the last claim, $\widehat{A}-A$ is a column submatrix of $\widehat{B}-B$, so $\|\widehat{A}-A\| \le \varepsilon_B$, and Weyl's inequality for singular values gives $s_{\min}(\widehat{A}) \ge s_{\min}(A)-\varepsilon_B \ge s_{\min}(A)/2 > 0$, hence $\|\widehat{A}^+\| = 1/s_{\min}(\widehat{A}) \le 2\kappa$. □

A.2.3 THE THEOREM

Theorem 2 (Affine readout, error uniform in hull distance up to a calibration term)**.** *Let Assumptions 5 to 8 hold and let $\mathcal{R}$ be the readout in (5). Fix $\eta,\eta' \in (0,1)$ with $\kappa\,\varepsilon_B(\eta') \le 1/2$. With probability at least $1-\eta'$ over the calibration contexts, the following nested statement holds: for every $\theta \in \Theta_{\mathrm{ev}}$, an independent evaluation context $c \sim D_\theta$ satisfies, with probability at least $1-\eta$,*

$$\big\|\mathcal{R}(c)-\theta\big\| \;\le\; 2\kappa\,\varepsilon_n(\eta) \;+\; 2\kappa\,\varepsilon_B(\eta')\big(1+R_H+d(\theta)\big). \tag{8}$$

Specifically:

(i) (population readout) *for $\mathcal{R}^\star(c) = A^+(\psi(c)-b)$, which is (5) with $\widehat{B}$ replaced by B, no calibration event is needed. For every $\theta \in \Theta_{\mathrm{ev}}$, the evaluation event gives*

$$\big\|\mathcal{R}^\star(c)-\theta\big\| \;\le\; \kappa\,\varepsilon_n(\eta) \;=\; O\Big(\kappa\,\sigma\sqrt{(m+\log(1/\eta))/n}\Big), \tag{9}$$

which does not contain $d(\theta)$. For fixed σ, it is the same at every task covered by the assumptions.

(ii) (fitted readout) *the term $2\kappa\varepsilon_B(\eta')\,d(\theta)$ is the only one in (8) that grows with hull distance. The coefficient of $d(\theta)$ in this upper bound is $2\kappa\varepsilon_B(\eta') = O\big(\kappa\sigma(nNK\gamma)^{-1/2}\big)$. With the other quantities fixed, it tends to zero as the number K of independent calibration contexts grows. This is a coefficient in an error bound, not a bound on an empirical regression slope.*

Proof. Work on the calibration event of Lemma 5, which does not depend on θ. On it, $\widehat{A}$ has full column rank, so $\widehat{A}^+\widehat{A} = I_d$, and $\|\widehat{A}^+\| \le 2\kappa$.

Fix $\theta \in \Theta_{\mathrm{ev}}$ and work on the evaluation event of Lemma 4, which has the stated probability and is independent of the calibration event by Assumption 8. Write $u = (\theta,1)$ and $\bar{\xi} = \psi(c) - Bu$, so $\|\bar{\xi}\| \le \varepsilon_n(\eta)$. Using $\psi(c) = Bu+\bar{\xi}$ and $\widehat{B}u = \widehat{A}\theta+\hat{b}$,

$$\begin{aligned}\mathcal{R}(c)-\theta &= \widehat{A}^+\big(Bu+\bar{\xi}-\hat{b}\big)-\theta \\ &= \widehat{A}^+\big(Bu-\widehat{B}u+\widehat{A}\theta+\bar{\xi}\big)-\theta \;=\; \widehat{A}^+\big[\,\bar{\xi}+(B-\widehat{B})u\,\big],\end{aligned} \tag{10}$$

the last step by $\widehat{A}^+\widehat{A}\theta = \theta$. Taking norms, $\|\mathcal{R}(c)-\theta\| \le \|\widehat{A}^+\|(\|\bar{\xi}\|+\|\widehat{B}-B\|\|u\|) \le 2\kappa(\varepsilon_n(\eta)+\varepsilon_B(\eta')\|u\|)$. It remains to bound $\|u\|$. The hull is nonempty, closed and convex, so $\|\theta\| \le \|P_H(\theta)\|+\|\theta-P_H(\theta)\| \le R_H+d(\theta)$ by Lemma 1(ii), and $\|u\| = \sqrt{\|\theta\|^2+1} \le 1+\|\theta\| \le 1+R_H+d(\theta)$. This gives (8). For (i), repeat the display with $\widehat{B}=B$, where $\widehat{A}^+ = A^+$ has norm exactly κ and the second term vanishes; (i) needs no calibration event at all. Claim (ii) reads the rate off (7). For a finite set of T independently sampled evaluation contexts, one per task, replace η by η/T and take a union bound. Conditional on the calibration event, all T evaluation bounds then hold with probability at least $1-\eta$; jointly over calibration and evaluation, they hold with probability at least $1-\eta'-\eta$. The resulting change in the bound is logarithmic in T. □

Comparing the two bounds. Only the population readout has a bound with no $d(\theta)$ term. Define $E_{\mathrm{conf}}(\theta) = \|\theta-\hat{\theta}(x_0)\|$ and $E_{\mathrm{aff}}(\theta) = \|\mathcal{R}(c)-\theta\|$. For the fitted readout, define $a_0 = 2\kappa\varepsilon_n(\eta) + 2\kappa\varepsilon_B(\eta')(1+R_H)$ and $b_0 = 2\kappa\varepsilon_B(\eta')$. If the confined readout also has proportional slack with

$s_{\text{conf}} = 1 - \rho > 0$, the two theorems give $E_{\text{conf}}(\theta) \geq s_{\text{conf}} d(\theta)$ and $E_{\text{aff}}(\theta) \leq a_0 + b_0 d(\theta)$ on their stated events. These envelopes are strictly separated only if

$$b_0 < s_{\text{conf}} \qquad \text{and} \qquad d(\theta) > \frac{a_0}{s_{\text{conf}} - b_0}. \tag{11}$$

The coefficient condition alone is insufficient because the affine upper bound has a positive intercept. These pointwise envelopes also do not bound free-intercept regression slopes over a band. We do not estimate κ, σ, γ, ρ, or the resulting threshold, so Equation (11) is not an empirical claim about our curves.

Where the confinement went. On the event of Theorem 2(i), $\|\mathcal{R}^\star(c) - \theta\| \leq \kappa \varepsilon_n(\eta)$, so for any evaluation task with $d(\theta) > \kappa \varepsilon_n(\eta)$ the output lies strictly outside H. By Lemma 1(i) no convex combination of the training parameters ever does, so $\mathcal{R}^\star$ is not of the form assumed in Assumption 1, and the same holds for $\mathcal{R}$ whenever the right side of (8) is smaller than $d(\theta)$. Theorems 1 and 2 are therefore not in conflict: they describe different readout classes, and the hull constraint belongs to the first class alone. The general move of breaking convex aggregation to escape a hull is due to Yan et al. (2026b) in supervised set-to-scalar regression, where the break is a learned rescaling of the residual around the mean embedding. Here the break is a fixed affine map of a designed statistic, and what escapes is the task-parameter hull rather than a token-embedding hull.

Remark 3 (What Theorem 2 does not give). Eight limits, stated plainly. *(1) The statistic is designed by hand.* Assumption 5 is supplied by an analyst who knows the family. The theorem is about what a readout class can express, not about what a learner will find on its own. It is an existence proof about routing, not a deployable method; a deployed agent would have to discover ϕ. *(2) The bound is on the estimate, not on behavior.* Nothing here bounds return. A correct task code helps only if the policy is competent at codes it was not trained on; the oracle-input experiment of Section 7 shows an empirical case in which that competence fails. On HalfCheetah-Vel above the hull, the statsC probe tracks the parameter at $+0.933 \pm 0.169$ while its achieved-speed slope is -3.930 ± 0.963. This contrast is empirical and is not an application of the theorem. *(3) The noise level is assumed flat.* Assumption 6 uses one σ across Θ_{ev}. If prompts get noisier off the hull, or the prompt-collection policy degrades there, then σ is itself a function of $d(\theta)$ and the bound picks up hull dependence quietly through σ. *(4) The affine identity is itself an extrapolation claim.* Equation (4) is checked on training tasks and asserted beyond them. If the family changes qualitatively off the hull, for instance a new contact mode or a saturating actuator, the identity fails and the theorem claims nothing. *(5) The prompt-collection policy must not change.* A and b are properties of the pair (family, prompt distribution). Off-hull prompts drawn from a different behavior distribution can change both. *(6) The independence condition excludes our trajectory prompts.* The proof applies to independent transitions, whereas the experiments use contiguous five-step trajectory segments. Extending the result to dependent prompts needs a separate concentration argument and verified dependence constants. *(7) It says nothing about the trained transformer.* The learned statsC module, its jointly trained trunk, and its FiLM parameters are not $\mathcal{R}$. Whether a model can learn the constructed inverse is a separate question. *(8) The constants can be large, and flatness is exact only for the population readout.* The fitted bound retains a term linear in $d(\theta)$, and the full separation condition is Equation (11). None of its required constants is estimated here. We do not claim a bound that is constant in hull distance for a readout fit from finite data.

Remark 4 (Fit the readout forward, not backward). The readout in (5) regresses the statistic on the known training parameters and then inverts. The other order, regressing θ on ψ across training tasks, looks more direct but generally does not recover A^+. After centering both variables, suppose the parameter has covariance Σ_θ, the statistic noise is independent, with covariance Σ_e, and the inverse below exists. The population coefficient of the backward fit is $\Sigma_\theta A^\top (A \Sigma_\theta A^\top + \Sigma_e)^{-1}$. In the scalar isotropic case this is the familiar attenuation toward the training mean; in general it is a covariance-dependent bias, not necessarily a Euclidean contraction. The forward fit avoids errors-in-variables bias because the known parameters form the design. Our diagnostic scripts use backward regression, and statsC learns a separate map through the action objective, so neither implements the readout analyzed in this theorem.

B TASK FAMILIES, SPLITS AND THE OFFLINE DATASETS

This appendix settles three questions that the main text uses rather than argues: what a task is in each family, how a held-out task is made, and where the data every agent was trained on came from.

We will give the fourteen families and their task parameters first, then the splits and the distance from each band to the training hull, then the procedure that collected the raw rollouts, then what the converted datasets contain and how large they are, and finally the prompt bank, the contexts and the widened family. Task coordinates and declared bands are taken from the dataset manifests, which also record each parameter formula and the environment source checked. For Table 5, we recompute H from the 20 listed training coordinates and project each held-out coordinate onto that finite hull. The archived MetaWorld run manifests use a legacy field called `hull` for the generator's nominal design box; it is not used as H. The corrected exporter names that field `design_box` and writes the finite training points under `hull`.

B.1 Families and task parameters

Each family is a set of 40 tasks that share one state space, one action space and one horizon, and differ in a single low-dimensional task parameter θ. This is a hidden-parameter MDP (Doshi-Velez & Konidaris, 2016) widened so that θ may enter the reward as well as the dynamics, since most of our families vary a target speed, a heading or a goal position rather than a mass or friction multiplier. Eight families are MuJoCo locomotion (Todorov et al., 2012) and six are MetaWorld manipulation (Yu et al., 2020), and they divide into three kinds by what the parameter changes. A reward-shift family varies what the agent is asked to do, a dynamics family varies the body it does it with, and a goal-box family varies where a target sits. The parameter of each family, the grid it is drawn from and its dimensions are in Table 4.

Table 4: The fourteen families, with the task parameter, its dimension d_θ, the observation and action dimensions, and the rule that generates the 40 task parameters. Every family has an episode cap of 200 steps.

Family	task parameter θ	d_θ	$\lvert S\rvert$	$\lvert A\rvert$	grid over the 40 tasks
Reward-shift families (MuJoCo)					
HalfCheetah-Vel	target forward speed (m/s)	1	20	6	$\theta_i = 0.1 + 0.1\,i$
Ant-Dir	target heading (rad)	1	27	8	$\theta_i = i\pi/20$
Humanoid-Dir	target heading (rad)	1	376	17	$\theta_i = i\pi/20$
Ant-Goal	goal position (x, y)	2	29	8	two sampled semicircles, see text
Dynamics families (MuJoCo)					
Hopper-Mass	$\log_{1.5}$ body mass multiplier	1	11	3	geometric, 1.5^{-2} to 1.5^{2}
Walker-Mass	$\log_{1.5}$ body mass multiplier	1	17	6	geometric, 1.5^{-2} to 1.5^{2}
Hopper-Friction	$\log_{1.5}$ geom friction multiplier	1	11	3	geometric, 1.5^{-2} to 1.5^{2}
Walker-Friction	$\log_{1.5}$ geom friction multiplier	1	17	6	geometric, 1.5^{-2} to 1.5^{2}
Goal-box families (MetaWorld), band axis in brackets					
mw-reach	goal (x, y, z) $[z]$	3	36	4	farthest-point grid per band
mw-reach-wall	goal (x, y, z) $[z]$	3	36	4	farthest-point grid per band
mw-push	goal (x, y, z) $[x]$	3	36	4	farthest-point grid per band
mw-soccer	goal (x, y, z) $[x]$	3	36	4	farthest-point grid per band
mw-pick-place	goal (x, y, z) $[z]$	3	36	4	farthest-point grid per band
mw-plate-slide	goal (x, y, z) $[x]$	3	36	4	farthest-point grid per band

For the four dynamics families, the analysis parameter θ is $\log_{1.5}$ of the physical multiplier. All reported hull distances and slopes use that coordinate.

In the MetaWorld families the object start pose is frozen at its nominal value and only the goal moves, so the task parameter is the goal alone, and the observation the agent sees at the first step is identical across the tasks of a family. Ant-Goal is the one two-dimensional family: the twenty training goals are $(r\cos a, r\sin a)$ for $r \in \{0.8, 1.2\}$ and ten angles evenly spaced in $[-\pi/2, \pi/2]$, the ten moderate goals keep those angles at $r = 1.6$, and the ten extreme goals keep $r = 1.6$ and rotate past the edge of the sector. Note that Ant-Dir and Humanoid-Dir have a circular parameter, so a task far enough past one edge of the training sector is also close to the other edge, and a signed displacement from the hull stops being well defined; their extreme band reaches exactly that point, which is why we read their moderate band and nothing else.

Ant-Goal has one evaluation-grid duplication: moderate task 29 and extreme task 30 both have parameter $(0, 1.6)$ because both angle grids include $\pi/2$. They were collected as separate task IDs but describe the same MDP parameter. The reported extreme aggregates retain the declared ten-task split for traceability and therefore include this duplicate; we treat them as supporting results and base the main Ant-Goal interpretation on the moderate band.

B.2 Splits and the distance from each band to the hull

For every family, every held-out parameter has strictly positive distance from the convex hull of the 20 training parameters; Table 5 reports the ranges. The moderate and extreme evaluation labels are also disjoint except for one Ant-Goal endpoint described below. This is the property that the headline protocols of these papers lack, since their main splits resample held-out tasks from inside the training range: Prompt-DT interleaves five test velocity indices among thirty-five training ones on Cheetah-vel (Xu et al., 2022, Table 5), Meta-DT draws its forty-five training and five test velocities from one $U[0.075, 3.0]$ (Wang et al., 2024, App. B.1). For its main continuous-control families, T2MIR samples training and evaluation parameters from the same stated generator, using $45/5$ splits for Point-Robot, Cheetah-Vel, Ant-Dir, Walker-Param, Reach, and Push; DarkRoom instead uses an $80/20$ grid split (Wu et al., 2025, App. D). DPT holds out twenty goals of the same 10×10 grid it trains on (Lee et al., 2023, App. A.4), and Algorithm Distillation calls its Dark Key-to-Door evaluation in-distribution generalization over unseen tasks of the same generator (Laskin et al., 2023). Where these papers do step outside the range, in the Ant-dir experiment of Xu et al. (2022, §6.4), the Ant-Dir appendix of Wang et al. (2024, App. G), the Cheetah-Vel appendix of Wu et al. (2025, App. H.1) and the adversarial bandit of Laskin et al. (2023, §5), they score by return or regret alone and never read the task estimate. We use the train/test-context vocabulary surveyed by Kirk et al. (2023) and add the hull distinction. Every family has 20 training, 10 moderate and 10 extreme tasks, and Table 5 gives the hull, the two bands and the distance from each band to the nearest point of the hull.

Table 5: The training hull and two held-out generator bands. For MetaWorld, brackets show the declared band on the shifted axis, while parentheses show the full Euclidean distance in goal space to the convex hull of the 20 sampled training goals. The dynamics families are listed in $\log_{1.5}$ units.

Family (units)	training hull	moderate band (distance)	extreme band (distance)
HalfCheetah-Vel (m/s)	$[1.1, 3.0]$	$[0.6, 1.0] \cup [3.1, 3.5]$ (0.1 to 0.5)	$[0.1, 0.5] \cup [3.6, 4.0]$ (0.6 to 1.0)
Ant-Dir (rad)	$[1.571, 4.555]$	0.157 to 0.785 past each edge	0.942 to 1.571 past each edge
Humanoid-Dir (rad)	$[1.571, 4.555]$	0.157 to 0.785 past each edge	0.942 to 1.571 past each edge
Ant-Goal	convex polygon of the 20 points at $r \in \{0.8, 1.2\}$, $a \in [-\pi/2, \pi/2]$	$r = 1.6$, same sampled angles (0.4 radial)	$r = 1.6$, rotated past the angular extent (0.40 to 0.88)
Hopper/Walker-Mass (log)	$[-0.974, 0.974]$	$\pm[1.077, 1.487]$ (0.10 to 0.51)	$\pm[1.590, 2.000]$ (0.62 to 1.03)
Hopper/Walker-Friction (log)	$[-0.974, 0.974]$	$\pm[1.077, 1.487]$ (0.10 to 0.51)	$\pm[1.590, 2.000]$ (0.62 to 1.03)
mw-reach (z, m)	convex hull of 20 sampled goals	$[0.135, 0.185]$ (0.01094 to 0.06917)	$[0.275, 0.300]$ (dropped; 0.15092 to 0.18189)
mw-reach-wall (z, m)	convex hull of 20 sampled goals	$[0.135, 0.185]$ (0.01087 to 0.06775)	$[0.275, 0.300]$ (dropped; 0.15088 to 0.17823)
mw-push (x, m)	convex hull of 20 sampled goals	$[-0.040, 0.010]$ (0.01036 to 0.06071)	$[0.075, 0.100]$ (dropped; 0.12565 to 0.15041)
mw-soccer (x, m)	convex hull of 20 sampled goals	$[0.028, 0.055]$ (0.08323 to 0.11074)	$[0.075, 0.100]$ (dropped; 0.13065 to 0.15531)
mw-pick-place (z, m)	convex hull of 20 sampled goals	$[0.100, 0.150]$ (0.01077 to 0.06478)	$[0.265, 0.300]$ (dropped; 0.17604 to 0.21172)
mw-plate-slide (x, m)	convex hull of 20 sampled goals	$[-0.045, 0.025]$ (0.01037 to 0.08043)	$[0.075, 0.100]$ (dropped; 0.13064 to 0.15521)

For each MetaWorld row, the bracketed band is only the dominant-axis range. The reported distance is the three-dimensional Euclidean projection distance from each held-out goal to the convex hull of the 20 exact training goals; it is not the gap to the nominal train-band box.

In the two rows written in $\log_{1.5}$ units, $\pm[a, b]$ means the band occupies $[a, b]$ on both sides of zero, since a mass or a friction multiplier can be scaled up or down. Two entries of that table are listed

and then not used. The extreme band of the two circular families is the wrap case described above. The MetaWorld extreme band was excluded from agent-result analysis under a decision fixed before any model was evaluated there; its coordinates and corrected post hoc model-free diagnostics are retained only for audit. Thus, OOD means the moderate band throughout the reported MetaWorld agent results.

The MetaWorld bands are deliberately close to the nominal train-band box, which sets the generator's intended shift but is not the finite-task convex hull. The mean displacement of the moderate band beyond that box, along the band axis, is 0.0366 m on mw-reach and mw-reach-wall, 0.0360 m on mw-push, 0.0971 m on mw-soccer, 0.0366 m on mw-pick-place and 0.0465 m on mw-plate-slide, which is 0.73, 0.73, 0.72, 1.39, 0.52, and 0.66 success radii, respectively. The actual Euclidean distances to the sampled-task hull are in Table 5 and are slightly larger for some goals. The SAC expert solves every one of these held-out tasks, showing that they remain feasible even though the tested agents can lose performance there.

B.3 How the offline data were collected

The offline data are not the replay buffer of one multi-task agent. For every one of the forty tasks of every family we trained a separate SAC agent (Haarnoja et al., 2018) on that task alone, sharing no parameters and no data across tasks, and recorded evaluation rollouts at snapshots taken along its own training curve. Note that a buffer built this way is a mixture of behavior qualities rather than a set of expert demonstrations, since the early snapshots come from a policy that has not solved the task yet. This is analogous to the mixed-policy datasets included in D4RL (Fu et al., 2020) and is the property that makes the return token of a decision transformer worth conditioning on.

Each snapshot contributes twenty episodes from the deterministic actor, which is the tanh-squashed mean of the Gaussian policy, and fifty from the stochastic one, so a retained snapshot is worth seventy trajectories of at most two hundred steps. Snapshots are taken every fifty thousand environment steps on the MuJoCo families and every twelve and a half or twenty five thousand on MetaWorld, where the policies converge sooner, and collection runs to one million environment steps on the MuJoCo families other than Ant-Goal, which runs to three million, and to five hundred thousand on mw-reach. Ant-Goal is also the one family collected with the stochastic actor alone, which is why its deterministic count in Table 6 is zero and why its prompt bank falls back to the highest-return stochastic trajectories. Each actor is a two-hidden-layer network of width 1024 with ReLU activations that emits a Gaussian over actions with log standard deviation clipped to $[-5, 2]$, and it reads the raw environment observation, since the collection wrapper rescales actions into $[-1, 1]$ and leaves observations untouched.

Two properties of this procedure matter for what we read out of it later. The held-out tasks were collected exactly as the training tasks were, by the same script with the same budget and the same snapshot grid, so the in-hull and out-of-hull buffers use the same collection protocol. This controls a procedural source of the behavior-policy shift targeted by Yuan & Lu (2022), although the induced behavior distributions can still differ across tasks. The per-task actors are kept rather than discarded: they are the expert bank that action inversion compares an emitted action against in Appendix D, and every number this paper calls the SAC expert is one of these agents evaluated on its own task.

B.4 What the datasets contain, and how large they are

The conversion keeps a window of late snapshots per family rather than everything on disk, and the window is the one the classical context encoders of Appendix C were trained on, since the comparison in Section 5 is meant to be one protocol rather than two. On HalfCheetah-Vel that window runs from one hundred thousand steps to nine hundred and fifty thousand, which is eighteen snapshots and therefore 1260 trajectories per task, and on Ant-Goal it runs from one million to two million nine hundred and fifty thousand, which is forty snapshots of fifty stochastic episodes. Throughout Table 6 the per-task count is seventy times the number of retained snapshots, Ant-Goal excepted, where it is fifty times.

The fourteen families together hold a little over a million trajectories and about 42 GB, of which Humanoid-Dir alone is more than a quarter, since its observation is 376-dimensional. Where two counts are given in a row, a few tasks of that family were collected with extra snapshots and carry

Table 6: The converted offline datasets, one row per family, after the snapshot window is applied. The deterministic column counts the rollouts the prompt bank draws from, and the last column is the per-family constant that normalizes the return token.

Family	traj / task	deterministic / task	total traj	size (GB)	rtg scale
HalfCheetah-Vel	1260	360	50 400	1.24	500
Ant-Dir	1260	360	50 400	1.38	1300
Ant-Goal	2000	0	80 000	2.68	2400
Humanoid-Dir	1330	380	53 200	12.28	1200
Hopper-Mass	1260	360	50 400	0.74	800
Walker-Mass	1260	360	50 400	1.04	1100
Hopper-Friction	1330 or 4130	380 or 1180	64 400	0.83	800
Walker-Friction	1330 or 2730	380 or 780	56 000	1.11	1100
mw-reach	420	120	16 800	0.60	2000
mw-reach-wall	2030 or 3010	580 or 860	82 180	2.95	1900
mw-push	1680	480	67 200	2.41	1800
mw-soccer	3570	1020	142 800	5.12	1800
mw-pick-place	4830 or 7350	1380 or 2100	195 720	7.02	1800
mw-plate-slide	2100	600	84 000	3.01	1800

the larger one, which is also why the total is not always forty times the per-task count. Trajectories are capped at 200 steps, and the families whose episode can end on a fall, hopper, walker and the two ant families, also contain shorter ones.

B.5 Conversion to the sequence format

The raw rollouts are stored as arrays of per-step tuples, which is the wrong shape for a sequence model. So a conversion pass writes one file per task holding a list of trajectories with observations, actions, rewards, terminal and timeout flags, timesteps, and the return-to-go, the last of these being the suffix sums of the raw rewards. Note that the conversion changes no value: actions are kept as the collection wrapper produced them, already in $[-1, 1]$, rewards are never rescaled, and the stored next observation is dropped rather than carried, since it is the following step's observation and keeping it would double the size for nothing. A terminal flag is set only where the episode genuinely ended early, which happens when the body falls in the hopper, walker and ant families, while a step that ends at the two hundred step cap is marked as a timeout instead. Consequently, an agent trained on these files never reads a time limit as a failure.

Three things travel with every trajectory: the task index, the task parameter θ, and a provenance string naming the snapshot and the rollout it came from, so any trajectory in any batch can be traced back to the file it was read out of. Each family also carries a manifest with the observation and action dimensions, the episode cap, the split assignment, the snapshot window, the return-to-go scale that normalizes the return token, and the observation mean and standard deviation. Those statistics are computed over training tasks and the retained window alone, and the same constants are then applied to every task at evaluation, which means a held-out task never contributes to a normalizer it is later measured against.

The task parameters are not read out of the rollouts, which never contain them: for the MuJoCo families they are regenerated from the environment's own deterministic formula, and for the MetaWorld families they are read from the family manifest that fixed the forty goals, whose object coordinates are held at their nominal value so that the goal alone is the parameter. Two checks stand behind the conversion, and both are cheap enough to rerun on any family: we checked on sampled trajectories that the dropped next observation equals the following observation, which is what makes dropping it lossless, and that the summed rewards of a converted trajectory equal the return recorded for that episode in the collection log at the same snapshot.

B.6 Prompts and contexts

The prompt bank holds, per task, the three highest-return deterministic trajectories, and the few-shot prompt is a contiguous five-step segment of the best of them. That prompt is fixed rather than sampled: the same one is used during training evaluation, during rollout logging and by every instrument, so no result in this paper depends on prompt sampling noise. The zero-shot protocol passes no prompt at all. mini-DPT instead receives a context of $H = 100$ transitions drawn i.i.d. from the evaluation task's own offline buffer under a fixed per-task seed, and its self-context protocol starts from an empty context and appends the agent's own transitions as it goes. Note that the few-shot context of an out-of-hull task therefore contains that task's own data by design, which is the point of the design: the question is whether the agent uses information it demonstrably has, and Appendix G audits that the information is in fact there.

B.7 The widened family

The widening experiment adds ten virtual training tasks at $\theta = 3.05, 3.10, \ldots, 3.50$ to HalfCheetah-Vel, built by exact reward relabeling of trajectories from tasks at or below $\theta = 3.0$, which is the relabeling idea of Dorfman et al. (2021) used as an intervention rather than as a method. The original forty tasks are hardlinked byte for byte, so the control and the widened arm share their data exactly outside the ten added tasks. The far band, $\theta \in [3.6, 4.0]$, was never a relabeling source and never enters training, which is what makes it the only band still legitimately outside the widened hull. Appendix G gives the identity check that makes the relabeling exact.

C Implementation details for every arm

This appendix settles whether the arms differ in anything other than the path the prompt takes on its way to the policy. We will describe the control first, then mini-DPT, then the two replacement arms together with the checks that establish what their masking does and does not do, and finally the classical encoders and Meta-DT, both of which are evaluated on existing checkpoints rather than trained here.

C.1 Prompt-DT, the control

Prompt-DT (Xu et al., 2022) prepends a trajectory prompt to a Decision Transformer stream (Chen et al., 2021) of returns-to-go, states and actions, and lets full cross-episode attention aggregate the prompt. Our implementation is a fork of the authors' model with the GPT-2 backbone replaced by a self-contained vendored copy, so that no result depends on a library version, and its hyperparameters are in Table 7. This arm is the control everywhere in the paper, since every other decoder-only arm shares its backbone size, its data, its optimizer budget and its evaluation protocol.

C.2 mini-DPT

mini-DPT follows the Decision-Pretrained Transformer (Lee et al., 2023), in which a query state token is followed by H context transition tokens (s, a, s', r) and the model is trained to predict the expert action at the query state. Our adaptation to continuous control changes four things from the reference implementation, and we list them because each one matters for comparability. The action head is linear under an MSE loss rather than a classifier; a LayerNorm is applied after the transition embedding; and every context length from 0 to H is supervised rather than only the non-empty ones, since the self-context protocol starts empty. Positional embeddings are removed, because the context is an unordered set of transitions, which also makes the hidden-state indexing identical to the Prompt-DT arm and therefore makes the probe measure the same thing on both. Labels are the deterministic action of that task's SAC actor at the query state, cached to disk, and two checks say those labels are what we think they are. Replaying an expert rollout through the batched relabeling path reproduces the emitted actions to at most 9.8×10^{-7} on five tasks. Labelling one task's states with all twenty training actors identifies the own actor by minimum error on five tasks out of five, so the labels are task-discriminative, which is what the supervision requires.

C.3 THE STATSC AND FILMB ARMS

Both replacement arms keep the prompt tokens in the sequence, in the same positions and under the same causal mask, and zero out the attention from the main stream to the prompt block. Therefore the only remaining route from the prompt to the policy is a code $z \in \mathbb{R}^8$ produced by a small encoder over the prompt transitions and injected by per-block FiLM modulation (Perez et al., 2018). The two arms differ in the encoder alone. filmB uses a single-layer GRU with mean pooling over valid prompt steps, which is a learned aggregator behind the same bottleneck, while statsC uses $z = W\phi + b$, an affine map of a hand-designed per-family summary ϕ, which is a sum-of-embeddings readout in the sense of Zaheer et al. (2017) rather than a learned pool; Appendix E gives every ϕ. The FiLM parameters are initialized from a small normal rather than at zero, since an exactly zero initialization makes the gradient into the encoder vanish at the first step and the encoder never starts, which we measured as encoder gradient norms of 0.0000 under zero initialization against 0.375 and 0.034 under the small-normal one.

Three checks establish that the masking does what it claims, and each is exact , not statistical: holding z fixed and changing the prompt content leaves the output exactly unchanged, and the gradient on the prompt embedding parameters is exactly zero. The default control path is bit-identical to the pre-intervention trainer over 300 iterations of loss values in full double precision and over all 67 tensors of the final checkpoint, checked twice on different machines. However, the two replacement arms do carry about 26k extra parameters in the encoder and the modulation heads, which sit outside the backbone: the backbone is identical and the total is slightly larger, and we say so rather than claiming an exact parameter match.

C.4 CLASSICAL CONTEXT ENCODERS

The classical column is evaluation-only, on existing checkpoints of three offline meta-RL context encoders, CSRO (Gao et al., 2023), UNICORN (Li et al., 2024) and ER-TRL (Nakhaeinezhadfard et al., 2025), three seeds each, on HalfCheetah-Vel and Ant-Goal. These methods descend from the FOCAL and PEARL line (Li et al., 2021; Rakelly et al., 2019), where CSRO and its predecessors optimize variants of a single mutual-information objective between the task and its representation (Li et al., 2024) and ER-TRL regularizes the mutual information between that representation and the behavior policy (Nakhaeinezhadfard et al., 2025), which is why one conclusion about the family of methods is meaningful at all. For every task we infer the code z from a noise-free offline context, roll out the z-conditioned deterministic policy with the code held fixed, and fit a ridge readout from z to θ on training tasks only, with the regularization chosen by leave-one-task-out cross-validation at the task level. Returns are raw environment rewards and are never rescaled, and the training-time Bellman reward scale is recorded in the run metadata and applied to no reported number.

C.5 META-DT

Meta-DT (Wang et al., 2024) is included as a case study and nothing more, as its training-task attainment does not clear the line we fixed in Appendix D, so it never enters a performance comparison with Prompt-DT; the details and the reason for the demotion are in Appendix G, and it has three seeds rather than six, since seeds beyond its third were never trained.

C.6 WHAT WAS HELD FIXED, AND THE SEEDS

Equal optimizer steps at equal batch size means the arms see the same number of supervised pairs, and the seeds are six per family and architecture for the grid, the swap and the oracle decomposition, with four blocks at three that we name wherever they appear: the widened training range, the prompt-composition arms, the classical encoders, and Meta-DT, whose later seeds were never trained. Every $\pm$ in this paper is an across-seed standard deviation of a band mean unless the text around it says otherwise, we never mix it with a per-task standard deviation, and the one table that uses a different convention says so where it is discussed.

Table 7: Hyperparameters. The upper block is shared by every decoder-only arm trained here, and the lower block gives what each arm changes.

Shared by Prompt-DT, mini-DPT, statsC, filmB, and the prompt-composition arms	
embedding width	128
attention heads	1
dropout, activation	0.1, ReLU
batch size	128
learning rate, weight decay	10^{-4}, 10^{-4}
warmup steps	10 000
gradient clip	0.25
training iterations	100 000
evaluation	40 tasks, 5 episodes per task, every 10k iterations
seeds	0 to 5, with four blocks at 0, 1, 2
Per-arm differences	
Prompt-DT (control)	3 layers, context $K = 20$, prompt length 5, full cross-episode attention
statsC	3 layers, $K = 20$, prompt length 5, prompt attention masked, $z = W\phi + b$, $z \in \mathbb{R}^8$
filmB	as statsC, with a single-layer GRU encoder of hidden width 64 in place of $W\phi + b$
mini-DPT	4 layers, context $H = 100$ i.i.d. transitions, linear action head, no positional embedding
prompt-composition arms	Prompt-DT with the prompt drawn i.i.d. from the buffer, $P = 5$ or $P = 100$

D INSTRUMENTS

This appendix settles what each instrument measures, and where it can be trusted. Three of them decode the implicit task estimate and one is a control: a probe reads the estimate out of the hidden state, inversion reads it out of the emitted actions, and a behavioral readout reads what the body actually did, with the layer-0 probe printed beside the probe as an input baseline. We will describe the three in turn, then the checks that qualified each behavioral index, then the attainment lines that decide which instrument may be quoted on which family, and finally the statistical conventions that hold everywhere in the paper.

D.1 ROLLOUT-STATE RIDGE PROBE

The probe is a ridge regression $\hat{\theta} = Wh + b$ from the residual stream to the task parameter, in the tradition of linear probes for continuous latents (Gurnee & Tegmark, 2024) and of probing for latent world state (Li et al., 2023). We follow the probing discipline that Hewitt & Liang (2019) argue for, namely that a probe's accuracy should reflect the representation not what the probe itself can fit: the probe is fitted on training tasks only and reported next to a control. Their control is a random-label control task and ours is a control on the representation, the layer-0 probe of the same rollouts, so we borrow the discipline rather than the instrument. For the rollout-state probes, hidden states are logged while the agent acts, at the current state token of the main sequence, at every environment step and at all $n_{\text{layer}} + 1$ depths. The one exception is the explicitly marked Ant-Goal offline probe. Its segment states already contain the achieved expert behavior. The layer-0 task-level cross-validated R^2 is 0.647, and the selected headline layer reaches 0.920. We report that diagnostic only as contaminated descriptive evidence, not as an online-code estimate.

The readout is fitted on training-task timesteps alone, so moderate and extreme timesteps are pure evaluation and are never seen by the fit. The regularization is chosen per layer over the grid $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 100\}$ by five-fold cross-validation split at the task level rather than at the timestep level, so that no fold shares a task with its own training part, and the headline layer is the layer with the best training-task cross-validated R^2 among layers at depth one or greater. Layer 0, the post-embedding input to the first block, is reported beside it as the control. Note that the headline layer differs by seed on HalfCheetah-Vel, layers $1, 1, 3$ and $1, 3, 3$ over the three seeds of the two arms of the widening experiment, with cross-validated R^2 differing by at most 0.006 across the candidate layers, so nothing in the paper turns on which of them is picked.

The ridge target lives in the family's analysis space, matching the inversion: $\log_{1.5} \theta$ for the dynamics families, whose grid is geometric, and $(\cos\theta, \sin\theta)$ for the circular families, whose angle is not a linear target and whose estimate is decoded back by a circular mean. For mini-DPT the layer-0 control is read from the context token stream, since the layer 0 of its readout token is the last context transition embedding and is an in-context audit quantity rather than an input baseline.

Two probe quantities are never quoted anywhere in this paper. The first being the R^2 pooled over an evaluation split. This stays high even on a completely saturated band, because that band sits at both ends of the hull and the between-band variance then dominates, so what it reports is the geometry of the split rather than the quality of the probe. Only the training-task cross-validated R^2 is citable. The second is any per-task number, since a task carries five evaluation episodes and nothing more.

D.2 Action inversion

Inversion compares the action the agent emits against the deterministic actions of all forty per-task SAC experts at the same state. The hard estimate is the parameter of the nearest expert, $\hat{\theta}(t) = \theta_{\arg\min_j \|a_t - a_j^*(s_t)\|_2}$, and the soft estimate is the softmax-weighted average of the forty parameters with weights $\propto \exp(-\|a_t - a_j^*(s_t)\|^2/\tau)$, where the temperature τ is selected on training-task timesteps by minimum squared error to the true parameter and is therefore never tuned on a held-out task.

Two failure modes are handled explicitly, not hoped away. When the selected τ reaches the top of its grid the soft posterior has collapsed to the prior mean and the soft estimate is not a reading of anything. Nine of the secondary-protocol arms are in that state, their soft estimates are excluded, and the conclusions there are carried by hard inversion, the probe and the returns. The few-shot arms, by contrast, are all safe, with selected τ between 0.14 and 4.11. The second failure mode is the expert bank itself, so before any estimate is computed we roll the task-zero expert actor in the real environment and require its return to match the tail of that task's recorded learning curve, which catches a mismatched or misloaded bank early.

D.3 Behavioral readouts and how each index was checked

The behavioral readout asks what the body did, with no reference to the hidden state, and every index was checked against expert data before it was used. On HalfCheetah-Vel the readout is the achieved forward speed, observation index 8, averaged over the steady phase $t \geq 100$; the index was picked by correlating each observation dimension's per-task mean with the target speed on deterministic expert data, and it matches the documented observation layout. On Ant-Dir the readout is the heading of the mean velocity vector, observation indices 13 and 14, whose correlations with $\cos\theta$ and $\sin\theta$ on expert data are $+0.916$ and $+0.992$, and the decoded heading is 2.6 degrees from the true heading on expert rollouts. However, the same construction on Humanoid-Dir gives a mean angular error of 121.5 degrees, which is not a reliable readout, so that family is excluded from every conclusion and its numbers appear only as observations.

On MetaWorld the readout is a band-axis statistic: the hand height over the five steps best aligned to the goal in the horizontal plane for the two reach families, the mean object x over $t \geq 150$ for push, soccer and plate-slide, and the mean object z over $t \geq 150$ for pick-place. Two earlier candidates were rejected on expert data. A plain late-window mean fails on reach because the hand hovers at a goal-independent height after passing through the target. Aa maximum-excursion statistic, which fails since the initial state itself sits inside the moderate band. The surviving readout still correlates with the goal on expert data only between 0.607 and 0.898, well short of the 0.99 the cheetah speed readout reaches, so it is never read against an identity line. The headline MetaWorld readout is therefore a signed gap to the expert on the same statistic. Appendix F additionally reports axis-only gains relative to the nominal generator-box edge, always labeled as behavioral quantities and, where interpreted, shown with the SAC-expert reference. The expert-relative comparison controls for a common scale and offset, although it need not remove every policy-dependent measurement bias.

D.4 Attainment lines and which instruments pass

An instrument that cannot recover the parameter where the answer is known cannot be trusted where it is not, so we fixed two lines on training tasks: inversion top-1 identification at least 0.70, and probe training-task cross-validated R^2 at least 0.70. Table 8 applies both lines to all fourteen families and both architectures, with the layer-0 control printed beside every probe value, since a high control value weakens the probe next to it rather than supporting it.

Table 8: Instrument quality on training tasks, few-shot, six seeds. Bold marks a decoder that clears the 0.70 attainment line, and the layer-0 columns are the input control and are never marked.

	Prompt-DT			mini-DPT			
Family	inv top-1	probe R^2	layer 0	inv top-1	probe R^2	layer 0	verdict
HalfCheetah-Vel	**0.909**	**0.961**	0.897	**0.933**	**0.919**	0.836	both pass
Ant-Goal	0.559	**0.920**	0.647	**0.908**	0.607	0.422	one decoder each
Ant-Dir	0.661	**0.829**	0.619	0.692	**0.727**	0.595	probe carries both
Humanoid-Dir	0.148	0.461	0.190	0.224	0.555	0.005	instrument-limited
Hopper-Mass	0.182	0.140	−0.009	0.212	0.697	−0.062	instrument-limited
Walker-Mass	0.407	0.287	0.149	0.472	0.444	0.017	instrument-limited
Hopper-Friction	0.193	0.199	0.252	0.206	0.505	0.161	instrument-limited
Walker-Friction	0.322	−0.061	0.000	0.459	−0.274	−0.046	instrument-limited
mw-reach	0.230	−0.143	0.311	0.453	0.462	−0.281	readout carries it
mw-reach-wall	0.179	0.459	0.437	0.498	0.407	−1.053	readout carries it
mw-push	0.305	−0.640	−0.156	0.427	0.173	0.094	readout carries it
mw-soccer	0.142	−0.364	−0.273	0.284	−0.628	−0.284	readout carries it
mw-pick-place	0.320	0.568	0.459	0.659	−0.156	0.366	readout carries it
mw-plate-slide	0.200	0.435	0.320	0.685	0.337	0.220	readout carries it

The verdict column names the evidence each family is then read with. Three families out of fourteen have a decoder that clears its line, and they are the three the main text draws decoding conclusions from. Ant-Goal misses the inversion line on Prompt-DT and the probe line on mini-DPT, and Ant-Dir misses the inversion line on both, in each case by a margin with a structural explanation. With forty candidate experts on a dense grid, inversion is dominated by confusion between neighboring tasks, so its top-1 rate understates how well the emitted action identifies the parameter. On Ant-Dir, we read inversion with the passing probe for both agents. On Ant-Goal, we read Prompt-DT inversion only with its passing but contaminated probe, while the mini-DPT result rests on its passing inversion rather than its failing probe. The remaining eleven are instrument-limited, their held-out-band instrument quantities are appendix observations, not measured snap. For the six MetaWorld families the citable evidence is not decoding at all but the success rate and the behavioral readout gap of Section 5. Note that on the four mass and friction families the band means of Appendix F sit near the origin rather than at the hull edge, which says the estimate does not track the parameter anywhere, in the hull or out of it, and that is a failure to read θ, as opposed to a snap.

D.5 Statistical conventions

Each task is evaluated with five episodes, so per-task quantities are noisy and are never reported: every number in this paper is a band mean or a split mean, and every $\pm$ is the standard deviation across the six seeds of that band mean. Four blocks stay at three seeds and say so in their captions: the widened training range, the prompt-composition arms, the classical encoders and Meta-DT, whose seeds beyond the third were never trained. One further exception is the Ant-Goal column of the manipulation table, where the aggregator's convention is the mean of per-task across-seed standard deviations, 0.188 and 0.189 against band-mean values of 0.066 and 0.072, and we say so there instead of mixing the two conventions silently.

E The hand-designed context statistics

This appendix settles what the statistic ϕ is in each family, how it was screened before any arm was trained, and why the corrected test rejects six of the fourteen families. We will state first what the experiment is and what it is not, then give every ϕ, then the pre-check and its corrections, and finally the exclusions and what they say about the families they remove.

E.1 What this experiment is, and what it is not

The intervention replaces learned prompt aggregation with an affine map of a statistic designed from each family's reward and observation layout. It shows how a supplied summary can change a separately trained agent, but it changes the conditioning architecture, its parameters, and optimization. It is neither a deployable method nor the forward-fit inverse analyzed in Theorem 2. Designing ϕ also requires family knowledge, so we report the results as an exploratory architecture intervention.

E.2 The statistics, one family at a time

HalfCheetah-Vel. The reward is $-|v_t - \theta| - 0.05\|a_t\|^2$. On the collected near-expert prompts, achieved forward speed empirically tracks θ both in and out of range; we have not established unbiasedness or an independent-sample concentration rate. Here ϕ is the mean of observation index 8 over valid prompt steps, one-dimensional, denormalized with the training-task statistics stored in the checkpoint.

Ant-Dir. The reward is velocity projected on the target heading, so we use the two-dimensional mean of observation indices 13 and 14. The scalar angle is a nonlinear function of this velocity vector. Therefore this statistic does not satisfy Theorem 2 under the paper's scalar-angle parameterization; it is an empirical feature choice for the intervention.

Ant-Goal. The reward is negative distance to the goal plus control and contact terms, and the torso position is observed directly, so ϕ is the reward-weighted position over the prompt, two-dimensional, with weights $\mathrm{softmax}(r_t/\tau)$ at $\tau = 0.5$ chosen against the reward range of the converted expert data. On an expert prompt those weights concentrate on the transitions closest to the goal, and the plain mean is the $\tau \to \infty$ limit of the same statistic. Because these normalized weights depend on the whole context, this statistic is not the arithmetic mean of a fixed per-transition feature assumed by Theorem 2.

Humanoid-Dir. The same circular construction as Ant-Dir, on observation indices 219 and 224, chosen by a leave-one-task-out search over the 376-dimensional observation using training tasks only. The corrected pre-check rejects this statistic; an already completed run is reported but excluded, as detailed in Appendix F.

MetaWorld. The goal is three-dimensional and the reward is a saturating function of the distance to it. Here ϕ is the mean of selected observation coordinates: indices $(18, 19, 20)$ for the two reach families, $(4, 5, 6)$ for pick-place, $(4, 5, 28)$ for soccer and $(5, 22, 23)$ for plate-slide, all chosen by the same training-only search. The candidate for mw-push, indices $(7, 23, 25)$, is the one that fails the held-out-band pre-check below.

Mass and friction. The parameter is not recorded directly in the observation, so the candidate statistic is a mean state response over the prompt, on indices chosen by the same training-only search. These four families fail the pre-check; this rules out the tested summaries, not every possible statistic.

E.3 The statistic screen and its retrospective correction

Statistics were screened before a single GPU hour was spent on the arms. The procedure computes ϕ on each task's deterministic evaluation prompt, fits the affine map on training tasks only, and evaluates it on the held-out bands. Thus the OOD bands affect statistic and family selection even though they do not enter gradient training; the resulting intervention is exploratory. A later audit found implementation and criterion errors, so Table 9 reports a corrected post hoc screen rather than the exact rule that selected every completed run. An early version scored the aggregate relative error over all coordinates, which dilutes the main coordinate along which a band moves and wrongly rejected mw-reach. The corrected multidimensional rule selects the coordinate with the largest absolute difference between the pooled held-out mean and the training mean; a scalar family has only one choice. Its crossing rate is the fraction of held-out tasks whose estimate crosses the corresponding endpoint of the training tasks' axis projection in the direction of the true coordinate. A task whose true selected coordinate lies inside that projection counts as not crossing. A corrected pass also requires, among tasks whose true selected coordinate is outside, the mean absolute coordinate error divided task by task by the distance beyond that projection to be below 1.0; circular families

use the shortest angular error. An early version also computed ϕ over whole trajectories rather than over the five-step window the model actually sees. Finally, circular predictions were wrapped onto a branch that split the training sector, and a task that did not cross either selected-axis endpoint was mistakenly counted as a successful crossing. Both implementation errors are fixed in the reported rates. Table 9 gives the corrected rates, and a rate of zero there is a statement about the statistic and the family rather than about any model, because the statistic itself uses no model outputs.

Table 9: Corrected model-free screen of the hand-designed statistics. The original screen preceded arm training; these corrected rates were computed post hoc. The rate is the fraction of held-out tasks whose fitted statistic-based estimate crosses the corresponding endpoint of the selected coordinate projection of the finite training-task hull in the direction of the true coordinate. Crossing that supporting plane certifies leaving the hull, but the rate is a one-axis sufficient test and is not a Euclidean hull-distance measure. Bold marks only crossing rates of at least 0.80; the outcome also applies the normalized-error condition stated in the text and distinguishes the corrected verdict from a retained historical run.

Family	statistic ϕ	moderate	extreme	outcome
HalfCheetah-Vel	mean forward speed	**1.00**	**1.00**	kept
Ant-Dir	mean velocity vector	**0.90**	**0.80**	passes moderate
Ant-Goal	reward-weighted position	0.50	**1.00**	marginal; retained run
Humanoid-Dir	mean of two observation dims	0.10	0.00	excluded
mw-reach	mean selected obs. coordinates	**0.80**	**1.00**	kept
mw-reach-wall	mean selected obs. coordinates	**0.90**	**1.00**	kept
mw-pick-place	mean selected obs. coordinates	**0.80**	**1.00**	kept
mw-soccer	mean selected obs. coordinates	**1.00**	**1.00**	kept
mw-plate-slide	mean selected obs. coordinates	**0.80**	**1.00**	kept
mw-push	mean selected obs. coordinates	0.00	0.00	excluded
Hopper-Mass, Walker-Mass	mean state response	0.00	0.00	excluded
Hopper-Friction, Walker-Friction	mean state response	0.00	0.00	excluded

E.4 The six exclusions

Humanoid-Dir. Correctly unwrapping the circular prediction gives crossing rates of 0.10 and 0.00, so this statistic fails the pre-check. The completed intervention run is retained only for traceability and supports no conclusion.

mw-push. Its selected-coordinate statistic has a crossing rate of zero in both bands. This result rejects the tested summary for this family; it does not reject the family or every possible statistic.

The four mass and friction families. The parameter is not recorded directly in the observation and is identifiable only through the response to actions. We tried mean state responses, per-action response ratios, several dimension combinations, and prompt lengths of 20, 50 and 100 steps including whole trajectories, and every crossing rate stayed at zero. Increasing the tested window length did not rescue these summaries, suggesting limited excitation in the expert prompt. These negative tests show a limit of the statistics we tried, not an impossibility result for identifying dynamics parameters from expert demonstrations. The four families lie outside the intervention and remain instrument-limited observations.

An earlier version of the criterion excluded MetaWorld wholesale, and the reason it did is worth keeping. A statistic used to estimate a goal position in a family whose success criterion is a radius need not, and in these prompts did not, resolve positions much more precisely than that radius; the mean moderate-band target excesses are less than one and a half radii outside the nominal train-band box, so the aggregate criterion could not separate a statistic that resolves the displacement from one that does not. The corrected selected-coordinate criterion shows that five of the six families still cross reliably, while mw-push does not, for the separate reason above.

F Full per-family results

This appendix settles what every family does, including the eleven whose decoders miss the attainment lines of Appendix D. Those rows are marked and they are observations rather than measured snap, since reading a slope from an instrument that cannot recover the parameter on training tasks

would be reading the instrument rather than the agent. We will give the held-out-band instrument quantities first, then the band means behind them, then the layer-0 control that every probe row is printed against, then returns by split and the second protocol on each architecture, and finally the parts of the manipulation that the main text has no room for.

F.1 Instrument responses beyond the hull

Table 10 gives instrument quantities for thirteen families and both architectures. The MetaWorld entries are axis-only behavioral readouts; the other entries are decoded estimates. Soccer is omitted because its in-range band-axis slope has a seed standard deviation of 0.78. Other than Ant-Goal and MetaWorld, each held-out slope is read against the in-hull slope to its left: the pair expected under a hard snap is a slope near one inside the hull and near zero outside it. A zero held-out slope alone only says that the estimate is approximately constant; Table 11 checks whether that constant is near the boundary.

Table 10: Instrument quantities for thirteen families, few-shot, six seeds. Soccer is omitted because its in-range band-axis slope is unstable. Other than Ant-Goal and MetaWorld, slope one inside the hull with slope zero beyond it is the slope signature expected under a hard snap; the band means in Table 11 distinguish boundary pinning from a constant wrong estimate. For Ant-Goal, in-hull entries are multivariate OLS slopes and held-out entries are mean nearest-task components q. Bold marks, within a family and a side, the architecture whose instrument moves further. For MetaWorld, the in-hull entries are band-axis behavioral slopes and the upper entries are displacement gains relative to the nominal generator-box edge; neither is a hull-distance estimate.

	Prompt-DT			mini-DPT		
Family	in hull	lower	upper	in hull	lower	upper
HalfCheetah-Vel	$+1.008 \pm 0.004$	$+0.000 \pm 0.066$	$+0.151 \pm 0.031$	$+1.021 \pm 0.007$	$\mathbf{+0.498 \pm 0.158}$	$\mathbf{+0.447 \pm 0.035}$
Ant-Goal†	$+0.816 \pm 0.048$	-0.457 ± 0.066	-0.337 ± 0.072	$+0.962 \pm 0.020$	$\mathbf{-0.273 \pm 0.059}$	$\mathbf{-0.166 \pm 0.055}$
Ant-Dir†	$+0.981 \pm 0.012$	$\mathbf{+0.374 \pm 0.087}$	-0.159 ± 0.096	$+0.991 \pm 0.005$	$+0.266 \pm 0.108$	$\mathbf{+0.062 \pm 0.015}$
Humanoid-Dir*	$+0.649 \pm 0.060$	$+0.245 \pm 0.401$	-0.605 ± 0.371	$+0.748 \pm 0.057$	$+0.017 \pm 0.328$	$+0.491 \pm 0.181$
Hopper-Mass*	$+0.069 \pm 0.022$	$+0.020 \pm 0.022$	$+0.046 \pm 0.038$	$+0.113 \pm 0.025$	$+0.040 \pm 0.036$	-0.039 ± 0.042
Walker-Mass*	$+0.160 \pm 0.031$	$+0.012 \pm 0.017$	-0.029 ± 0.075	$+0.113 \pm 0.035$	$+0.005 \pm 0.025$	$+0.089 \pm 0.027$
Hopper-Friction*	$+0.101 \pm 0.009$	-0.075 ± 0.045	$+0.015 \pm 0.041$	$+0.083 \pm 0.021$	-0.030 ± 0.026	-0.053 ± 0.061
Walker-Friction*	$+0.062 \pm 0.012$	$+0.028 \pm 0.019$	$+0.013 \pm 0.043$	$+0.136 \pm 0.055$	$+0.074 \pm 0.031$	$+0.146 \pm 0.044$
mw-reach*	$+1.149 \pm 0.023$	*n/a*	$+1.086 \pm 0.021$	$+1.019 \pm 0.148$	*n/a*	$+1.020 \pm 0.258$
mw-reach-wall*	$+0.913 \pm 0.232$	*n/a*	$+1.209 \pm 0.149$	$+0.393 \pm 0.193$	*n/a*	$+0.571 \pm 0.134$
mw-push*	$+0.960 \pm 0.303$	*n/a*	-0.386 ± 0.223	$+0.982 \pm 0.156$	*n/a*	$+0.350 \pm 0.204$
mw-pick-place*	$+1.806 \pm 0.148$	*n/a*	$+0.459 \pm 0.747$	$+1.894 \pm 0.088$	*n/a*	$+0.142 \pm 0.182$
mw-plate-slide*	$+1.266 \pm 0.201$	*n/a*	$+0.204 \pm 0.135$	$+1.215 \pm 0.074$	*n/a*	$+0.455 \pm 0.095$

Four conventions hold in that table and in the two that follow it. A family marked $*$ is instrument-limited, both of its decoders having missed their lines, and only the three unmarked or † families are marked in bold, since a slope read from a decoder that misses its line is not a result; a family marked † has at least one failing decoder cell but a complementary passing decoder, and each row is interpreted under the arm-specific caveat in Table 8. The circular families report their moderate band, for the wrap reason of Appendix B. Ant-Goal reports the nearest-task component q defined in Section 4, where zero means no recovered component and one means recovery of the true component along that direction; q is unbounded, and the statistic does not determine orthogonal error or hull membership. Its two columns are the moderate and the extreme band rather than two sides of a hull. The MetaWorld families have no lower side and their held-out-band column is a band displacement gain relative to the nominal generator-box edge rather than a slope or hull-distance ratio. mw-soccer is omitted here because the in-hull seed spread of its band-axis slope is 0.78 for Prompt-DT and 1.34 for mini-DPT, since the ball rolling back out of the goal, so only its gap and its success rate are read anywhere in the paper. Every cell in this table has six seeds.

We can see two different failures in the table rather than one. On the four mass and friction families the in-hull slope is already near zero, so nothing is saturating and the decoder simply never recovers the parameter. For the circular families we report no wing-wide and no extreme-band slope anywhere, since those quantities are inflated by an antipodal wrap at the two tasks at the ends of the circle and by the unwrapping convention that follows it. The inflation is identical in the control

and in the replacement arm, so a difference between arms read off them would be an artifact of the unwrapping rather than a property of either arm.

F.2 Where the estimate lands

For rows that are slopes, a rate can be small either because the estimate stopped or because it never started. Table 11 therefore gives the band means behind the response quantities, in each family's units.

Table 11: Decoded band means, few-shot, six seeds, with the true band mean in parentheses. Bold marks, within a family and a side, the architecture whose band mean sits closer to the true mean. For Ant-Goal, lower/upper means moderate/extreme and the entries are decoded radii; radius alone does not determine full-vector error or hull membership.

Family (axis)	PDT lower (true)	PDT upper (true)	DPT lower	DPT upper
HalfCheetah-Vel (m/s)	$+1.128 \pm 0.059$ ($+0.300$)	$+2.986 \pm 0.023$ ($+3.800$)	$\mathbf{+0.668 \pm 0.098}$	$\mathbf{+3.369 \pm 0.040}$
Ant-Goal (radius)	$+1.032 \pm 0.026$ ($+1.600$)	$+0.973 \pm 0.019$ ($+1.600$)	$\mathbf{+1.099 \pm 0.024}$	$\mathbf{+1.096 \pm 0.041}$
Ant-Dir (rad)	-1.497 ± 0.033 (-1.963)	$+1.197 \pm 0.037$ ($+1.963$)	$\mathbf{-1.572 \pm 0.023}$	$\mathbf{+1.477 \pm 0.044}$
Humanoid-Dir (rad)*	-0.210 ± 0.088 (-1.963)	$+0.351 \pm 0.106$ ($+1.963$)	-0.768 ± 0.091	$+1.301 \pm 0.082$
Hopper-Mass (log)*	-0.021 ± 0.013 (-1.795)	$+0.059 \pm 0.023$ ($+1.795$)	-0.139 ± 0.034	$+0.047 \pm 0.010$
Walker-Mass (log)*	-0.250 ± 0.046 (-1.795)	-0.038 ± 0.037 ($+1.795$)	-0.236 ± 0.043	$+0.009 \pm 0.014$
Hopper-Friction (log)*	-0.077 ± 0.033 (-1.795)	$+0.022 \pm 0.020$ ($+1.795$)	-0.123 ± 0.029	-0.013 ± 0.033
Walker-Friction (log)*	-0.049 ± 0.011 (-1.795)	-0.014 ± 0.017 ($+1.795$)	-0.137 ± 0.031	$+0.068 \pm 0.027$

The one-dimensional MuJoCo families report their far band and the circular families their moderate band. Ant-Goal's two columns are again the moderate and the extreme band, and Table 11 separately reports decoded radius, whose outer training-sample radius is 1.2. On HalfCheetah-Vel the far band asks for a mean of 3.800 and Prompt-DT answers 2.986 against a hull edge of 3.000, which is the cleanest single reading of the effect in the paper. On Ant-Dir the mini-DPT upper band mean is $+1.477$ against a sector edge of $+1.492$, inside the sector. On HalfCheetah-Vel, only mini-DPT's decoded band means leave the old hull, on both sides. On the four dynamics families the band means sit between -0.25 and $+0.07$, while the true band mean is ± 1.795. This is the pattern of an estimate that never leaves the middle of the training range, not one pinned against its edge. This pattern is the reason those families are read as instrument-limited, not snapped.

F.3 The layer-0 input control

A probe value means nothing on its own, since some of the parameter is already linearly readable from the input to the first block, so every probe row above is printed here against its own input control (Table 12). A larger value in this table weakens the probe row beside it rather than strengthening it.

Ant-Goal has no Prompt-DT entry here, since that arm's probe is the offline segment probe and its two-dimensional aggregator produces no layer-0 slope, only the cross-validated R^2 of 0.647 that appears in Table 8. Note that on HalfCheetah-Vel the control is high inside the hull, $+0.909$ against a probe slope of $+1.008$, and collapses outside it just as the probe does, so the input carries much of the parameter in the hull and neither the input nor the probe carries it beyond.

F.4 Return by split

Return is the quantity that does not depend on any instrument, so we give it for every family in Table 13, with the SAC expert evaluated on the same splits as the reference. The expert column is a reference and never a competitor: on MetaWorld and on the dynamics families it is nearly flat across the splits, which is what says the drop belongs to the model rather than to the task. The MetaWorld success rates and readout gaps that the main text reads in place of a decoder are in Section 5, and are not repeated here. MetaWorld has no analyzed extreme split under our protocol, which is why it is a second panel rather than six more rows.

Table 12: Layer-0 input-control quantities, few-shot, six seeds. Ordinary scalar and circular rows are slopes; Ant-Goal held-out entries are nearest-task components q; MetaWorld in-hull entries are band-axis slopes and upper entries are nominal generator-box-edge displacement gains, not hull-distance estimates. Bold marks the stronger in-hull baseline, with both marked on a tie. The mini-DPT columns are read from the context-token stream.

	Prompt-DT			mini-DPT (context tokens)		
Family	in hull	lower	upper	in hull	lower	upper
HalfCheetah-Vel	$\mathbf{+0.909 \pm 0.003}$	$+0.029 \pm 0.064$	$+0.079 \pm 0.017$	$+0.862 \pm 0.006$	-0.150 ± 0.174	$+0.134 \pm 0.033$
Ant-Goal	*n/a*	*n/a*	*n/a*	*n/a*	-0.323 ± 0.057	-0.414 ± 0.068
Ant-Dir	$\mathbf{+0.860 \pm 0.004}$	$+0.373 \pm 0.082$	$+0.036 \pm 0.113$	$+0.858 \pm 0.012$	$+0.100 \pm 0.116$	$+0.065 \pm 0.090$
Humanoid-Dir	$+0.719 \pm 0.013$	-0.131 ± 0.457	-0.584 ± 0.394	$\mathbf{+0.728 \pm 0.021}$	-0.285 ± 0.194	$+0.114 \pm 0.234$
Hopper-Mass	$\mathbf{+0.144 \pm 0.050}$	$+0.164 \pm 0.119$	$+0.155 \pm 0.095$	$+0.083 \pm 0.022$	$+0.061 \pm 0.062$	-0.018 ± 0.049
Walker-Mass	$\mathbf{+0.338 \pm 0.070}$	$+0.095 \pm 0.105$	$+0.290 \pm 0.190$	$+0.225 \pm 0.029$	$+0.117 \pm 0.091$	$+0.128 \pm 0.071$
Hopper-Friction	$\mathbf{+0.324 \pm 0.037}$	-0.053 ± 0.128	-0.040 ± 0.136	$+0.229 \pm 0.042$	-0.018 ± 0.034	-0.094 ± 0.071
Walker-Friction	$\mathbf{+0.211 \pm 0.037}$	$+0.026 \pm 0.061$	$+0.047 \pm 0.056$	$+0.171 \pm 0.023$	$+0.036 \pm 0.061$	$+0.185 \pm 0.037$
mw-reach	$\mathbf{+0.594 \pm 0.026}$	*n/a*	-0.531 ± 0.043	$+0.363 \pm 0.073$	*n/a*	-0.526 ± 0.475
mw-reach-wall	$\mathbf{+0.600 \pm 0.056}$	*n/a*	-0.531 ± 0.096	$+0.274 \pm 0.073$	*n/a*	-0.914 ± 0.256
mw-push	$+0.478 \pm 0.055$	*n/a*	-0.676 ± 0.036	$\mathbf{+0.542 \pm 0.039}$	*n/a*	-0.517 ± 0.034
mw-pick-place	$\mathbf{+0.711 \pm 0.020}$	*n/a*	-0.212 ± 0.053	$\mathbf{+0.711 \pm 0.024}$	*n/a*	-0.237 ± 0.054
mw-plate-slide	$\mathbf{+0.555 \pm 0.034}$	*n/a*	-0.143 ± 0.021	$+0.483 \pm 0.037$	*n/a*	-0.173 ± 0.039

Table 13: Band-mean return by split, few-shot, six seeds. Bold marks the higher return of the two architectures within a split, and the expert column is the SAC expert on the same splits.

	ID		OOD moderate		OOD extreme		expert
Family	PDT	DPT	PDT	DPT	PDT	DPT	ID/mod/ext
HalfCheetah-Vel	-27.6 ± 0.7	$\mathbf{-26.5 \pm 1.2}$	-91.5 ± 2.2	$\mathbf{-52.3 \pm 6.1}$	-185.1 ± 6.0	$\mathbf{-104.0 \pm 6.4}$	$-22.1/-25.7/-31.9$
Ant-Goal	-96.0 ± 1.7	$\mathbf{-72.4 \pm 3.4}$	-180.4 ± 6.4	$\mathbf{-172.2 \pm 5.7}$	-264.8 ± 12.8	$\mathbf{-215.1 \pm 9.3}$	$-59.0/-78.0/-80.2$
Ant-Dir	877.1 ± 15.5	$\mathbf{897.6 \pm 14.8}$	638.0 ± 27.5	$\mathbf{682.3 \pm 34.6}$	85.7 ± 21.9	$\mathbf{239.8 \pm 39.3}$	902.5/863.0/941.5
Humanoid-Dir	675.2 ± 17.3	$\mathbf{806.9 \pm 24.1}$	663.8 ± 42.3	$\mathbf{848.2 \pm 53.1}$	647.2 ± 27.9	$\mathbf{793.1 \pm 42.7}$	997.7/1037.0/1055.5
Hopper-Mass	$\mathbf{552.0 \pm 24.6}$	454.3 ± 29.7	504.4 ± 44.2	$\mathbf{527.5 \pm 40.8}$	446.1 ± 28.0	$\mathbf{548.3 \pm 25.5}$	614.1/612.5/607.3
Walker-Mass	$\mathbf{685.6 \pm 19.1}$	668.4 ± 15.0	532.0 ± 63.2	$\mathbf{581.6 \pm 21.5}$	459.8 ± 83.0	$\mathbf{480.4 \pm 23.6}$	770.6/701.0/654.5
Hopper-Friction	$\mathbf{576.3 \pm 19.7}$	492.0 ± 44.4	435.4 ± 24.9	$\mathbf{474.1 \pm 24.3}$	376.2 ± 25.0	$\mathbf{391.9 \pm 44.2}$	621.4/587.4/470.4
Walker-Friction	$\mathbf{692.4 \pm 21.9}$	652.2 ± 21.9	595.5 ± 18.4	$\mathbf{651.1 \pm 62.3}$	579.8 ± 30.4	$\mathbf{628.5 \pm 47.3}$	803.9/771.0/775.8

	ID		OOD moderate		expert
Family	PDT	DPT	PDT	DPT	ID / moderate
mw-reach	$\mathbf{1830.6 \pm 5.8}$	1239.9 ± 68.7	$\mathbf{1598.3 \pm 38.3}$	1158.3 ± 187.0	1857.7 / 1856.3
mw-reach-wall	$\mathbf{1741.1 \pm 38.5}$	1380.7 ± 28.7	$\mathbf{1600.1 \pm 47.8}$	1101.9 ± 92.7	1836.8 / 1841.7
mw-push	$\mathbf{1473.1 \pm 47.4}$	1451.4 ± 67.3	$\mathbf{929.9 \pm 71.3}$	672.4 ± 122.9	1651.0 / 1653.3
mw-soccer	1095.7 ± 70.5	$\mathbf{1171.4 \pm 64.3}$	199.2 ± 76.7	$\mathbf{276.8 \pm 69.3}$	1573.5 / 1630.7
mw-pick-place	1430.4 ± 53.4	$\mathbf{1519.2 \pm 52.7}$	989.6 ± 89.5	$\mathbf{1268.0 \pm 60.5}$	1598.8 / 1589.6
mw-plate-slide	1656.4 ± 4.2	$\mathbf{1663.9 \pm 2.8}$	1265.7 ± 88.6	$\mathbf{1493.7 \pm 63.0}$	1678.2 / 1678.2

F.5 THE SECOND PROTOCOL ON EACH ARCHITECTURE

Removing the cross-episode channel removes the task inference, which for Prompt-DT means the empty prompt and for mini-DPT the self-context protocol, in which the agent builds its own context from scratch. The training-split returns under both are in Table 14, and the comparison there is about the channel rather than about performance: with no prompt, the dedicated task signal is absent, although the current state and dynamics may still contain task information, and return falls toward that of a single average policy.

An arm marked ‡ there is one whose soft-inversion temperature reached the top of its grid, so its posterior collapsed to the prior mean and its soft estimates are excluded everywhere in the paper. The two architectures fail differently under their secondary protocol. Prompt-DT with an empty prompt loses the in-hull slope entirely, so for that agent the prompt is the dominant measured channel. However, mini-DPT on its own context keeps a near-unit in-hull slope on HalfCheetah-Vel, $+1.053 \pm 0.039$ under soft inversion at a safe temperature, and loses the out-of-hull behavior instead. That cell misses one of the two attainment lines under this protocol, so we read it beside its return rather than alone. Together, these readings show that the cross-episode channel materially

Table 14: Training-split return under the primary and the secondary protocol, six seeds. Bold marks the better of the two protocols within an architecture.

	Prompt-DT		mini-DPT	
Family	few-shot	zero-shot	few-shot	self-context
HalfCheetah-Vel	$\mathbf{-27.63 \pm 0.73}$	$-132.22 \pm 18.86^{\ddagger}$	$\mathbf{-26.54 \pm 1.24}$	-45.80 ± 4.74
Ant-Goal	$\mathbf{-96.03 \pm 1.65}$	$-234.75 \pm 12.62^{\ddagger}$	$\mathbf{-72.37 \pm 3.41}$	-219.06 ± 2.86
Ant-Dir	$\mathbf{877.10 \pm 15.50}$	553.60 ± 16.60	$\mathbf{897.60 \pm 14.80}$	436.00 ± 41.40
Humanoid-Dir	675.20 ± 17.30	$\mathbf{792.10 \pm 76.40}$	$\mathbf{806.90 \pm 24.10}$	805.00 ± 74.00
Hopper-Mass	$\mathbf{552.00 \pm 24.60}$	$473.90 \pm 36.30^{\ddagger}$	454.30 ± 29.70	$\mathbf{582.00 \pm 12.90}$
Walker-Mass	$\mathbf{685.60 \pm 19.10}$	$313.40 \pm 43.30^{\ddagger}$	$\mathbf{668.40 \pm 15.00}$	547.50 ± 65.60
Hopper-Friction	$\mathbf{576.30 \pm 19.70}$	$442.90 \pm 75.10^{\ddagger}$	492.00 ± 44.40	$\mathbf{602.70 \pm 14.20}$
Walker-Friction	$\mathbf{692.40 \pm 21.90}$	$296.10 \pm 70.90^{\ddagger}$	652.20 ± 21.90	$\mathbf{797.80 \pm 59.80}^{\ddagger}$
mw-reach	$\mathbf{1830.60 \pm 5.80}$	1159.60 ± 132.80	$\mathbf{1239.90 \pm 68.70}$	933.20 ± 101.20
mw-reach-wall	$\mathbf{1741.10 \pm 38.50}$	1045.50 ± 425.10	1380.70 ± 28.70	$\mathbf{1555.40 \pm 59.80}$
mw-push	$\mathbf{1473.10 \pm 47.40}$	$364.70 \pm 300.90^{\ddagger}$	$\mathbf{1451.40 \pm 67.30}$	$607.80 \pm 340.60^{\ddagger}$
mw-soccer	$\mathbf{1095.70 \pm 70.50}$	512.30 ± 186.30	$\mathbf{1171.40 \pm 64.30}$	868.80 ± 224.80
mw-pick-place	$\mathbf{1430.40 \pm 53.40}$	322.50 ± 292.20	$\mathbf{1519.20 \pm 52.70}$	845.40 ± 295.00
mw-plate-slide	$\mathbf{1656.40 \pm 4.20}$	1323.10 ± 233.40	$\mathbf{1663.90 \pm 2.80}$	1379.00 ± 66.30

affects the estimate. They are consistent with, but do not prove, the hypothesis that its aggregation constrains whether the estimate leaves the hull.

F.6 The manipulation, in the parts the main text has no room for

The intervention itself, on the MuJoCo instruments and on MetaWorld success, is in Section 6. What is left for here is the band-axis readout under the same intervention, the Ant-Dir band means behind the slopes the main text quotes, the family we report and exclude, what the intervention does to the instruments themselves, and what it costs in return. Figure 4 summarizes these measurements: movement upward and right means that both reported response quantities increase, whereas movement mainly to the right means that only the representation-side quantity increases.

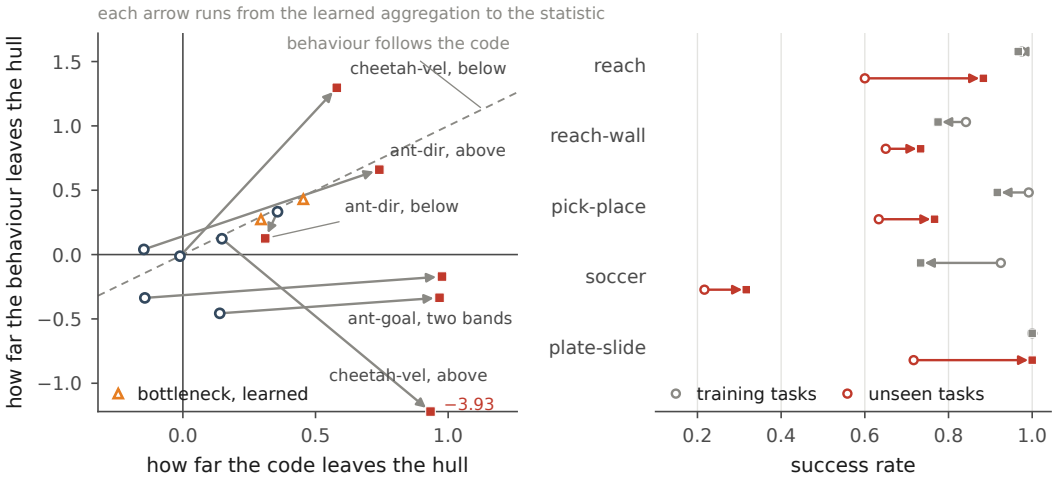

Figure 4: Each arrow compares the learned-aggregation agent with its separately trained statistic-routed variant. For HalfCheetah-Vel and Ant-Dir, both axes are OOD slopes; the horizontal axis is a rollout-state probe and the vertical axis is direct behavior. For Ant-Goal, both axes are nearest-task components q; the horizontal axis is the contaminated offline probe and the vertical axis is action inversion. Its arrow is descriptive only because the probe is contaminated and inversion misses its quality line in both arms. Neither axis is a hull distance.

The manipulation results. Table 15 gives the success rates and the readout gaps behind the two sentences the main text spends on MetaWorld, arm by arm and family by family. The gap columns are quoted to three decimals here, against four in Table 2, because the intervention aggregate reports them that way, and the control values for reach-wall come from a control rerun with success logging rather than from the sweep rollouts the main-text grid was built from.

Table 15: The architecture intervention, arm by arm, six seeds, with OOD the moderate band. Bold marks the higher success rate and the smaller absolute gap of each pair.

	training success		OOD success		OOD readout gap (m)	
Family	control	statsC	control	statsC	control	statsC
reach	**0.975±0.038**	0.967±0.037	0.600±0.082	**0.883±0.069**	−0.047±0.001	**−0.035±0.006**
reach-wall	**0.842±0.079**	0.775±0.138	0.650±0.050	**0.733±0.170**	−0.047±0.006	**−0.036±0.009**
pick-place	**0.992±0.019**	0.917±0.047	0.633±0.170	**0.767±0.125**	−0.029±0.027	**+0.004±0.039**
soccer	**0.925±0.063**	0.733±0.114	0.217±0.121	**0.317±0.134**	−0.155±0.076	**−0.096±0.036**
plate-slide	**1.000±0.000**	**1.000±0.000**	0.717±0.069	**1.000±0.000**	−0.048±0.006	**−0.034±0.013**

The band-axis readout on MetaWorld. Table 16 reads the same rollouts along each family's band axis, as the predicted axis displacement normalized by the true mean displacement beyond the nominal generator-box edge. This is an axis-only behavioral gain, not a Euclidean hull-distance ratio. Note that the reference here is the SAC expert on the same statistic and not the numeral 1, since the readout itself is soft, and no arm rises clearly above the expert: pick-place statsC reaches $+1.353$ against a reference of $+1.243$, a difference its seed spread of 1.055 does not resolve. Soccer is left out because its in-hull band-axis slope has a seed spread of 0.78.

Table 16: Band-axis behavioral gain relative to the nominal generator-box edge under the manipulation, few-shot, six seeds. Bold marks the larger of the two arms. The SAC expert on the same statistic is the reference and is never marked.

Family	control	statsC	SAC expert
reach	$+1.086 \pm 0.021$	$\mathbf{+1.413 \pm 0.173}$	$+2.377$
reach-wall	$+1.209 \pm 0.150$	$\mathbf{+1.505 \pm 0.236}$	$+2.492$
pick-place	$+0.460 \pm 0.746$	$\mathbf{+1.353 \pm 1.055}$	$+1.243$
plate-slide	$+0.204 \pm 0.135$	$\mathbf{+0.494 \pm 0.283}$	$+1.228$

Ant-Dir band means. The main text reports slopes, and the band means say the same thing in the units of the family. At a true moderate band mean of 1.9635 rad, with the sector edge at 1.4923, the control probe reads 1.196 ± 0.056, inside the edge, while the statsC probe reads 1.728 ± 0.050, outside it, and the achieved heading moves from 1.186 to 1.557. On this side the replacement arm walks past the boundary rather than only encoding past it.

Humanoid-Dir, reported and excluded. Table 17 gives the numbers, which are used in no conclusion. The corrected pre-check rejects the statistic; the earlier implementation had incorrectly classified it as marginal before this run was launched. It also fails the instrument lines on both arms, its heading readout is invalid at 121.5 degrees of expert angular error, and its few-shot return, 675.2, is below its zero-shot return, 792.1. These observations provide no evidence that the prompt improves task conditioning on this family.

Table 17: Humanoid-Dir under the manipulation, few-shot, six seeds. Reported for completeness and used in no conclusion, so nothing is marked, since none of these instruments clears its attainment line on this family.

Instrument	control lower	statsC lower	control upper	statsC upper	in hull (ctl / statsC)
soft action inversion	$+0.178 \pm 0.362$	$+0.530 \pm 0.695$	-0.626 ± 0.356	-0.227 ± 1.552	0.636 / 0.428
rollout-state probe	$+0.171 \pm 0.268$	$+0.974 \pm 0.527$	-0.780 ± 0.341	$+1.554 \pm 2.100$	0.961 / 0.774
layer-0 control	-0.131 ± 0.457	$+0.649 \pm 0.485$	-0.584 ± 0.394	-0.368 ± 0.586	0.719 / 0.608

Inversion top-1 on training tasks is 0.148 for the control and 0.102 for statsC there, and the probe's training-task cross-validated R^2 is 0.461 and -0.029, so the statsC probe values above one in that

table are not evidence of extrapolation: they come from a readout that does not work on training tasks.

Instrument quality under the manipulation. The replacement arm lowers the training-task inversion rate systematically, 0.909 to 0.561 on HalfCheetah-Vel, 0.661 to 0.478 on Ant-Dir, 0.559 to 0.435 on Ant-Goal and 0.148 to 0.102 on Humanoid-Dir. That drop is what we would expect of a policy whose prompt has passed through a bottleneck, since it is then less precisely matched to any one of forty experts, and its inversion slope is for that reason never quoted alone. The probe goes the other way and clears its line comfortably: training-task cross-validated R^2 of 0.961 against 0.994 on HalfCheetah-Vel, 0.829 against 0.907 on Ant-Dir and 0.920 against 0.993 for the Ant-Goal offline probe, with layer-0 controls of 0.897 against 0.909 and 0.619 against 0.625. The most visible consequence is the HalfCheetah-Vel upper band, where statsC inversion reads -3.607: that number reflects a policy that has stopped moving rather than a code that has gone backwards, which is why the probe and the achieved speed are what carry the claim there.

Returns under the manipulation. Table 18 gives band returns for both arms, and the replacement is plainly not a performance fix, since on the HalfCheetah-Vel far band it is much worse than the control. The family appears twice in that table because it has two legitimate controls, the standard Prompt-DT arm and the filmB arm, which shares the bottleneck and the masking of statsC and differs from it only in using a learned recurrent aggregator.

Table 18: Band-mean return, control against statsC, few-shot, six seeds. Bold marks the higher return in each pair. Humanoid-Dir is listed for completeness and is excluded from every conclusion.

	ID		OOD moderate		OOD extreme	
Family (control)	control	statsC	control	statsC	control	statsC
HalfCheetah-Vel (PDT)	-27.6 ± 0.7	$\mathbf{-26.0 \pm 0.4}$	-91.5 ± 2.2	$\mathbf{-83.5 \pm 8.9}$	$\mathbf{-185.1 \pm 6.0}$	-345.3 ± 61.6
HalfCheetah-Vel (filmB)	$\mathbf{-25.8 \pm 0.8}$	-26.0 ± 0.4	-93.7 ± 5.0	$\mathbf{-83.5 \pm 8.9}$	$\mathbf{-166.0 \pm 13.0}$	-345.3 ± 61.6
Ant-Dir	877.1 ± 15.5	$\mathbf{905.5 \pm 11.6}$	$\mathbf{638.0 \pm 27.5}$	598.2 ± 23.5	85.7 ± 21.9	$\mathbf{130.6 \pm 34.6}$
Ant-Goal	$\mathbf{-96.0 \pm 1.7}$	-99.2 ± 4.5	$\mathbf{-180.4 \pm 6.4}$	-188.1 ± 5.8	$\mathbf{-264.8 \pm 12.8}$	-278.8 ± 20.4
Humanoid-Dir (excluded)	675.2 ± 17.3	$\mathbf{712.9 \pm 21.8}$	$\mathbf{663.8 \pm 42.3}$	624.9 ± 23.6	$\mathbf{647.2 \pm 27.9}$	549.7 ± 15.2

Ant-Goal, a contaminated code probe. The offline probe's nearest-task component q is 0.967 ± 0.042 on the moderate band and 0.977 ± 0.053 on the extreme band, compared with control values of $+0.139$ and -0.143; its training-task cross-validated R^2 is 0.993. Action inversion stays negative, at -0.336 ± 0.251 and -0.171 ± 0.237, and return is slightly worse than the control. This ratio concerns one directional component and does not determine the full vector or hull membership. The probe uses offline expert segments rather than deployed-policy rollouts. That contamination can affect the two architectures differently, so neither its level nor the between-arm difference establishes the online task code. The extreme aggregate also includes the duplicated endpoint documented in Appendix B. We retain these numbers as descriptive evidence only.

G SUPPORTING EXPERIMENTS IN FULL

This appendix settles four questions that the main text answers in a sentence each. Is the failure beyond the hull one of inference or one of execution, and does the answer depend on which side of the hull we are on? Does the boundary move when the training range is widened, or does it dissolve? Is the out-of-hull task in the context at all, and does the way the context is composed change what the agent does with it? And do agents that build an explicit task code stop in the same place? We take them in that order, each subsection stating its design before its numbers, and the first two are the questions Section 7 answers in a sentence apiece.

G.1 HANDING THE POLICY AN OUT-OF-HULL ORACLE INPUT

This experiment separates two failures that look identical from the outside, since an agent may fail because it inferred the wrong task or because it cannot execute the right one. It runs on HalfCheetah-Vel, on the statsC arm, six seeds, forty tasks, five episodes each, few-shot.

The input substitution is algebraically exact. Analogous to task-vector interventions (Hendel et al., 2023), we replace the measured statistic in $z = W\phi + b$ by $\phi := \theta$ and reuse the checkpoint's own affine weights; the script checks this substitution with zero tolerance. We call the result the oracle-input code. The measured speed statistic tracks θ closely in our audit, but we have not proved that it is unbiased. Empirically, the inferred- and oracle-input conditions have similar mean returns: -26.05 versus -26.03 on training tasks, -82.39 versus -81.33 on the moderate bands, and -344.77 versus -343.76 on the extreme bands. The decomposition below therefore compares three interventions; it does not identify an exact latent estimate.

Table 19: Band-mean return on HalfCheetah-Vel under three conditions, six seeds, seed sd in parentheses. Bold marks the best of the three conditions in each band, and the expert column is the reference.

Band (θ)	expert	inferred	oracle	nearest-hull
training $[1.1, 3.0]$	-22.06	-26.05 (0.27)	$\mathbf{-26.03}$ (0.37)	$\mathbf{-26.03}$ (0.37)
moderate below $[0.6, 1.0]$	-4.79	-34.01 (8.00)	$\mathbf{-33.41}$ (7.72)	-66.33 (2.12)
extreme below $[0.1, 0.5]$	-1.87	$\mathbf{-58.63}$ (9.37)	-59.37 (9.20)	-162.61 (2.26)
moderate above $[3.1, 3.5]$	-46.52	-130.76 (14.54)	-129.26 (13.78)	$\mathbf{-100.43}$ (2.84)
extreme above $[3.6, 4.0]$	-61.97	-630.90 (116.42)	-628.14 (121.27)	$\mathbf{-200.48}$ (3.78)

The three conditions of Table 19 use the arm's measured statistic, the code produced by substituting the true parameter as its input, and the code produced from the nearest parameter inside the training range. Note that the two sides are never summed. Although the reward is symmetric in $|v - \theta|$, the learned policy responds asymmetrically: standing is close to a very low target, while the policy has limited high-speed behavior. Table 20 splits the gap between the expert and the nearest-boundary-input condition into the contribution of the oracle input and the gap that remains to the expert.

Table 20: Decomposition of the return gap on HalfCheetah-Vel, six seeds, seed sd in parentheses. Bold marks the larger of the two terms in each row, that is the binding constraint there. The two sides are never added.

Band	total gap	oracle-input contribution	remaining expert gap	recovered by oracle input
moderate below	61.53 (2.12)	$+\mathbf{32.92}$ (8.05)	28.61 (7.72)	53.4% (12.8)
extreme below	160.75 (2.26)	$+\mathbf{103.24}$ (10.62)	57.50 (9.20)	64.2% (6.0)
all below $[0.1, 1.0]$	111.14 (2.18)	$+\mathbf{68.08}$ (8.67)	43.06 (7.70)	61.2% (7.2)
moderate above	53.91 (2.84)	-28.83 (14.81)	$\mathbf{82.74}$ (13.78)	-54.3%
extreme above	138.51 (3.78)	-427.66 (120.12)	$\mathbf{566.17}$ (121.27)	-308.3%
all above $[3.1, 4.0]$	96.21 (3.18)	-228.24 (64.18)	$\mathbf{324.46}$ (64.58)	-237.3%

The total gap there is the expert minus the nearest-boundary-input condition, the oracle-input contribution is the oracle-input return minus that boundary return, and the remaining expert gap is the expert return minus the oracle-input return. Below the hull, the oracle-input condition recovers 61.2% of the gap and leaves 43.06 return points. Above the hull, the oracle input is harmful, which gives a negative contribution. On that side the boundary-input condition performs better than the oracle-input condition; this intervention comparison does not identify an exact latent estimate.

What the policy does out of the hull is neither divergence nor boundary tracking, and Table 21 shows it crouching to a stop instead.

Here v is the achieved speed, the two difference columns are against the target and against the nearest hull edge, and the residual is the part of the reward that the speed error does not explain. Boundary tracking would put v minus the edge near zero, which only the nearest-hull row does. Indeed the agent on the far band with the oracle-input code does not overshoot, does not ride the boundary and does not fall over. Its pitch is smaller than when it runs, and it simply lowers its body and stops, with a residual no larger than inside the hull. Therefore the return of -628 is a speed error of 2.997 sustained over two hundred steps, not an anomaly of the reward. The same environment yields -61.97 for the SAC expert in that band, so this is not a physical ceiling either.

Table 21: Steady-state behavior on HalfCheetah-Vel, $t \geq 100$, six seeds, band means. Bold marks the better reward per step in the far-above band, where the oracle-input and boundary-input codes are the two conditions being compared.

condition	band	v	$v - \theta$	$v-$ edge	torso height	|pitch| max	reward/step	residual
oracle	in hull	2.039	−0.011	−0.011	−0.119	0.233	−0.060	−0.012
oracle	below $[0.1, 1.0]$	0.503	−0.047	−0.597	−0.108	0.332	−0.234	−0.043
oracle	above $[3.1, 3.5]$	2.964	−0.336	−0.036	−0.120	0.195	−0.423	−0.063
oracle	above $[3.6, 4.0]$	0.803	−2.997	−2.197	−0.193	0.089	−3.045	−0.048
nearest-hull	above $[3.6, 4.0]$	2.988	−0.812	−0.012	−0.112	0.147	**−0.862**	−0.050

The oracle code might simply lie outside the range of modulation the network ever saw, so Table 22 gives its size against the range spanned by the inferred codes of the twenty training tasks, with the excess written as a multiple of the range width.

Table 22: Size of the FiLM modulation driven by the oracle code, six seeds, against the range spanned by the inferred codes of the twenty training tasks.

quantity	training range	below $[0.1, 1.0]$	above $[3.1, 3.5]$	above $[3.6, 4.0]$
$\|z\|$	$[2.606, 5.190]$	2.203 (0.15 out)	5.632 (0.17 out)	6.402 (0.47 out)
$\max \|\gamma\|$	$[1.398, 2.091]$	1.338 (0.16 out)	2.228 (0.19 out)	2.476 (0.53 out)
$\|\beta\|$	$[2.605, 3.872]$	3.345 (0.12 out)	4.310 (0.37 out)	5.119 (1.05 out)

The parenthetical in each cell is the mean seedwise excess beyond that seed's training range, divided by that range's width. Because this nonlinear summary is computed before averaging seeds, a mean value can fall inside the range formed from averaged endpoints, as in the below-hull $\|\beta\|$ cell. That table is a description and not causal evidence, since in this arm the code is affine in θ, so being far in parameter space and being far in code space are the same statement. Two falsifiable observations sit beside it: at the same far band an in-range code produces a policy that runs at 2.99, and the degradation grows gradually with the excess rather than at a cliff.

Two restrictions on this experiment are worth stating where they bite. It exists for HalfCheetah-Vel alone, because the substitution $\phi := \theta$ has a direct meaning only when the statistic and parameter use the same scalar units. The filmB arm has no usable oracle, since its code is produced by a learned recurrent aggregator, so the map from the parameter to the code has to be fitted rather than read off. That fit is poor enough that the resulting oracle is worse than the arm's own inference even on training tasks, so we report no oracle numbers for that arm, while its probe, inversion and return values are ordinary measurements and are used normally elsewhere.

G.2 Widening the training range

Does the boundary move with the training range, or does the confinement disappear altogether? The control is Prompt-DT trained on the original twenty tasks, $\theta \in [1.1, 3.0]$, three seeds. The treatment adds ten virtual training tasks at $\theta = 3.05$ to 3.50 in steps of 0.05, built by exact relabeling, three seeds, with bit-identical hyperparameters and only the training task set changed, and both arms are then measured on the same forty-task manifest with the same instruments. This is the intervention that the virtual-task line of work applies (Lee & Chung, 2021; Kim et al., 2025), alongside the related move of synthesizing pseudo-transitions inside the existing training tasks (Zhou et al., 2024), and a factor that studies of controlled pretraining task distributions find can affect in-context generalization (Raventós et al., 2023; Raparthy et al., 2024; Goddard et al., 2025).

The relabeling is exact rather than approximate, and we check it three ways. The reward identity $r_t = -|v_t - \theta| - 0.05\|a_t\|^2$ recovers v_t from a stored transition, and replaying those recovered speeds in the simulator reproduces the stored reward to a maximum absolute error of 2.4344×10^{-6} over 5373 steps, mean 5.8399×10^{-8}. A round trip through five virtual tasks whose parameter coincides exactly with a real task, 1260 trajectories each, reproduces that real task's stored rewards

to at most 1.9565×10^{-5} per step and its returns to at most 3.0518×10^{-5}. And the far band, task ids 35 to 39, was never used as a relabeling source and never enters training, which is what makes it the only legitimate out-of-hull test for the widened arm. We read the near band in Table 23, which the widening was meant to teach, and the far one in Table 24, which it was not.

Table 23: The near band, $\theta \in [3.1, 3.5]$, which is inside the widened training range and outside the control's, three seeds per arm, seed sd in parentheses. Bold marks the arm that treats the band as a task it knows.

Instrument	control value	control slope	widened value	widened slope
achieved speed	2.9315 (0.0078)	+0.5093 (0.0833)	**3.2832** (0.0092)	**+1.0236** (0.0492)
soft action inversion	2.9140 (0.0033)	+0.4310 (0.0696)	**3.2727** (0.0069)	**+0.9650** (0.0239)
hard action inversion	2.8836 (0.0051)	+0.2675 (0.0444)	**3.2688** (0.0044)	**+0.9746** (0.0314)
rollout-state probe	2.9069 (0.0097)	+0.4690 (0.0773)	**3.2025** (0.0907)	**+1.0972** (0.0075)
layer-0 control	2.8439 (0.0171)	+0.3746 (0.0652)	3.1082 (0.0091)	+0.7950 (0.0586)
return	−109.87 (1.47)	−107.45 (12.50)	**−54.28** (1.54)	**−10.45** (9.84)

The near band is a sanity check not a result, since if the widened arm had not learned the tasks it was given, nothing after it would mean anything, and the SAC expert returns -46.52 there. Bold there marks the arm whose value is closer to the true band mean of 3.3 and whose slope is closer to one, and on the return row the higher return. That band closes 0.9570 of the gap to the expert on the speed readout and 0.8584 on return, the two arms are almost identical inside the original hull, achieved speed 2.0479 against 2.0304 and return -27.04 against -29.16. The widened arm's instruments clear both attainment lines, with probe training-task cross-validated R^2 of 0.9582, 0.9602 and 0.9698 per seed against layer-0 controls of 0.9005, 0.8964 and 0.8903, and inversion top-1 of 0.902, 0.863 and 0.895.

Table 24: The far band, $\theta \in [3.6, 4.0]$, the only band the relabeling never touched, three seeds per arm. Bold marks the arm whose saturation value sits further beyond the old edge of 3.0. The expert return here is -61.97.

Instrument	control value	control slope	widened value	widened slope	far / in-hull
achieved speed	2.9809 (0.0063)	−0.1570 (0.0340)	**3.3268** (0.0147)	+0.1422 (0.0476)	0.1418
soft action inversion	2.9826 (0.0069)	−0.0512 (0.0187)	**3.3155** (0.0218)	+0.2509 (0.0522)	0.2473
hard action inversion	2.9577 (0.0122)	+0.0139 (0.0135)	**3.3126** (0.0283)	+0.2378 (0.0596)	*n/a*
rollout-state probe	2.9601 (0.0066)	−0.2225 (0.0647)	**3.2997** (0.1119)	+0.1677 (0.1524)	0.1695
layer-0 control	2.8757 (0.0108)	−0.1157 (0.0250)	3.1428 (0.0207)	+0.1244 (0.0643)	*n/a*
return	−200.04 (1.05)	−225.31 (4.61)	**−141.04** (3.14)	−172.99 (5.63)	*n/a*

In this experiment, widening the training range moves the saturation point instead of removing it. All four canonical instruments move their saturation value from between 2.958 and 2.983 in the control up to between 3.300 and 3.327 in the widened arm, just inside the new training edge of 3.5, and the layer-0 control moves with them, from 2.876 to 3.143. At the same time the far-band slope stays between 0.14 and 0.25 of the same arm's in-hull slope, which is the last column of the table, and the far band closes only 0.4277 of the return gap on the speed readout and 0.4327 on return. This result is consistent with a coverage-dependent boundary over the one range we could relabel exactly; it does not establish such a boundary for other ranges or families. One sentence we cannot write is that the widened arm clears the training-task attainment line, since its floor-normalized attainment was never computed.

G.3 Prompt composition, and whether the information is there at all

If the out-of-hull task is not reflected in the context, then a readout cannot recover it. This control checks one direct context statistic. Five arms, all on HalfCheetah-Vel, three seeds, 100k iterations, the same optimizer budget, the same instruments and the same protocol. Arm A is Prompt-DT with the trajectory prompt of five consecutive deterministic expert steps; arm B is Prompt-DT whose prompt is instead transitions drawn i.i.d. from the same task's whole offline dataset, at $P = 5$ and at $P = 100$; arm C is mini-DPT with an i.i.d. context, at $H = 100$ and at $H = 5$ as a budget

control. A against B changes composition with the architecture fixed, while B against C changes the architecture with composition nearly fixed.

Table 25: Beyond-hull slopes on HalfCheetah-Vel for five arms, three seeds. Bold marks the largest reported OOD slope in each row.

Instrument	A: prompt, traj $P = 5$	B: prompt, iid $P = 5$	B: prompt, iid $P = 100$	C: mini-DPT $H = 100$	C: mini-DPT $H = 5$
below the hull, $\theta < 1.1$					
soft inversion	-0.030 ± 0.069	$+0.051 \pm 0.023$	$+0.427 \pm 0.085$	$\mathbf{+0.561 \pm 0.171}$	$+0.543 \pm 0.118$
hard inversion	-0.041 ± 0.070	$+0.042 \pm 0.031$	$+0.483 \pm 0.089$	$+0.560 \pm 0.214$	$\mathbf{+0.634 \pm 0.191}$
probe (headline)	-0.028 ± 0.043	$+0.009 \pm 0.023$	$+0.401 \pm 0.033$	$\mathbf{+0.650 \pm 0.026}$	$+0.618 \pm 0.122$
layer-0 control	-0.000 ± 0.073	$+0.053 \pm 0.029$	$+0.219 \pm 0.061$	-0.290 ± 0.128	-0.222 ± 0.236
achieved speed	-0.039 ± 0.072	-0.006 ± 0.019	$+0.385 \pm 0.074$	$\mathbf{+0.871 \pm 0.226}$	$+0.821 \pm 0.324$
above the hull, $\theta > 3.0$					
soft inversion	$+0.150 \pm 0.021$	$+0.237 \pm 0.080$	$+0.132 \pm 0.033$	$\mathbf{+0.458 \pm 0.017}$	$+0.410 \pm 0.029$
hard inversion	$+0.146 \pm 0.029$	$+0.218 \pm 0.080$	$+0.129 \pm 0.037$	$\mathbf{+0.485 \pm 0.021}$	$+0.436 \pm 0.028$
probe (headline)	$+0.137 \pm 0.025$	$+0.273 \pm 0.054$	$+0.220 \pm 0.097$	$\mathbf{+0.568 \pm 0.235}$	$+0.357 \pm 0.005$
layer-0 control	$+0.085 \pm 0.016$	$+0.110 \pm 0.043$	$+0.063 \pm 0.037$	$\mathbf{+0.126 \pm 0.016}$	$+0.107 \pm 0.016$
achieved speed	$+0.117 \pm 0.017$	$+0.226 \pm 0.093$	$+0.172 \pm 0.030$	$\mathbf{+0.378 \pm 0.023}$	$+0.315 \pm 0.047$

All five arms track similarly inside the hull, with soft inversion between 0.991 and 1.019, headline probe between 0.977 and 0.992, layer-0 control between 0.858 and 0.913 and steady achieved speed between 1.002 and 1.039, so Table 25 is a comparison of what happens outside a hull that all five agents handle similarly inside. Below the hull the largest differences align with prompt composition, while above it the larger difference is between the two architectures. These comparisons are descriptive, not a factorial causal decomposition. Band returns follow the same pattern: on the training split the arms are within a few points of each other, -27.04 ± 0.38, -29.97 ± 0.54, -27.96 ± 0.22 and -26.00 ± 1.42 for A, B at $P = 5$, B at $P = 100$ and C at $H = 100$, they separate on the moderate bands, -92.99 ± 2.20, -86.64 ± 4.87, -78.31 ± 0.27 and -53.98 ± 8.14, and they separate further on the extreme bands, -188.68 ± 6.28, -176.44 ± 6.18, -156.78 ± 2.85 and -98.43 ± 3.17. Returns for the $H = 5$ budget control were not logged; as a result, that fifth arm is omitted from this return comparison.

The information audit is model-free, and it is the reason none of this can be explained by a context that says nothing about the task. We regress the raw readout of the context itself, the mean of the achieved-speed observation over the context transitions, on the true parameter, so that a value near one means the context carries the parameter one for one (Table 26).

Table 26: Model-free audit of the context, HalfCheetah-Vel. Each entry is the slope of the raw context readout against the true parameter, with no model involved. Bold marks the composition that carries the parameter best in each region.

Context composition	below the hull	in hull	above the hull
mini-DPT context, $H = 100$	$+0.987$	$+0.953$	$+0.882$
prompt i.i.d., $P = 100$	$+0.987$	$+0.953$	$+0.882$
prompt i.i.d., $P = 5$	$+0.888$	$+0.797$	$+0.816$
prompt trajectory, $P = 5$	$\mathbf{+1.008}$	$\mathbf{+1.007}$	$\mathbf{+1.043}$

Note that the trajectory prompt has the largest context-audit slope in every region but yields the smallest decoded OOD response of the five arms. The i.i.d. prompt at $P = 100$ and the mini-DPT context are identical in this audit, since the two are built by the same rule, while their soft-inversion slopes differ by a wide margin, $+0.427$ against $+0.561$ below the hull. Thus, lack of target-speed information in this measured statistic does not explain the observed pattern. The i.i.d. arms also learn the training tasks: under the floor-normalized attainment definition their seed means are 0.843 ± 0.015 at $P = 5$ and 0.885 ± 0.012 at $P = 100$ against 0.893 ± 0.011 for the reference Prompt-DT arm, with 50, 56 and 59 of 60 task-seed cells above the 0.80 line. All three therefore clear the line on the seed mean, and the slope differences cannot be attributed to an arm that failed to learn. That reference value of 0.893 is the variant computed with a corrected floor for one task,

the kill-test verdict elsewhere in this paper quotes 0.898 ± 0.009 for the same arm under the original floor, and the two are never mixed inside a single table.

G.4 Classical context encoders

Do agents that build an explicit task code stop in the same place? The classical column answers that on frozen checkpoints, under the same probe protocol as everything else, and Table 27 gives it in full. The in-hull slope is the trace of the multivariate ordinary least squares map over training tasks divided by the parameter dimension, the beyond-hull slope regresses the recovered excursion component on the true excursion, so a component that tracks exactly would have slope one, and the gain is the ratio of their sums, which stays defined when the slope degenerates. Neither quantity determines orthogonal error or hull membership. Ant-Goal's moderate band needs the gain, since every excursion there is radial by the same amount and the horizontal variance is zero.

Table 27: Classical context-encoder codes under the same probe protocol, three seeds. Bold marks the largest requested-direction response value per family. For Ant-Goal, the in-hull and pooled beyond-hull columns are slopes, while the moderate and extreme columns are gains.

Family	encoder	in hull	beyond hull	moderate	extreme
HalfCheetah-Vel	CSRO	0.9342 ± 0.0575	0.0668 ± 0.0055	0.2250 ± 0.0304	0.0200 ± 0.0067
HalfCheetah-Vel	UNICORN	0.9984 ± 0.0002	$\mathbf{0.0983 \pm 0.0389}$	$\mathbf{0.2252 \pm 0.0592}$	$\mathbf{0.0712 \pm 0.0316}$
HalfCheetah-Vel	ER-TRL	0.9703 ± 0.0079	0.0536 ± 0.0212	0.1805 ± 0.0437	0.0432 ± 0.0124
Ant-Goal	CSRO	0.9381 ± 0.0279	-0.0231 ± 0.0495	-0.3598 ± 0.0888	-0.3257 ± 0.0355
Ant-Goal	UNICORN	0.9898 ± 0.0098	-0.1206 ± 0.1949	$\mathbf{-0.1259 \pm 0.1110}$	$\mathbf{-0.1949 \pm 0.0862}$
Ant-Goal	ER-TRL	0.8980 ± 0.0288	$\mathbf{0.0275 \pm 0.1704}$	-0.2863 ± 0.0714	-0.1828 ± 0.0358

A unit recovered excursion component would be 1.0 in that table and no encoder is anywhere near it, the reconstruction-based encoder included. Its HalfCheetah-Vel empirical mean margin over the two contrastive encoders is small and was not tested for significance. In code space the out-of-hull codes usually lie near the outermost training clusters: the distance from an out-of-hull code to the nearest training code, divided by the median nearest-neighbor distance among the training codes, has medians of 0.0177, 0.2071 and 0.0213 on HalfCheetah-Vel for CSRO, UNICORN and ER-TRL and 0.2381, 0.4446 and 0.2134 on Ant-Goal. The maxima are 0.6966, 1.1042, 0.7348, 0.7295, 0.9379 and 0.8009, and a value below one means the code lands inside the spacing of the training clusters. The offline context contains task information through its rewards. These observations are consistent with compression at the encoder but do not rule out nonlinear encoding or decoder extrapolation error.

Band returns for the classical column, in the order training, moderate, extreme: on HalfCheetah-Vel, CSRO -43.47 ± 3.47, -108.47 ± 0.92, -195.11 ± 12.14; UNICORN -44.42 ± 2.93, -95.72 ± 4.96, -190.50 ± 7.12; ER-TRL -41.83 ± 1.69, -101.17 ± 2.30, -202.61 ± 2.24; on Ant-Goal, CSRO -114.00 ± 1.18, -221.51 ± 7.24, -290.59 ± 12.08; UNICORN -173.95 ± 56.65, -287.82 ± 75.72, -330.47 ± 77.33; ER-TRL -169.76 ± 46.18, -293.75 ± 55.92, -333.66 ± 38.47. The matched expert windows are -31.34, -35.08 and -43.12 on HalfCheetah-Vel and -125.82, -154.46 and -158.81 on Ant-Goal. Two quality flags travel with those numbers: one Ant-Goal ER-TRL seed has a probe cross-validated R^2 of 0.4687 against 0.8382 and 0.8573 for its siblings, and one seed each of Ant-Goal UNICORN and ER-TRL is a return outlier.

G.5 Meta-DT, a case study on the code channel

Meta-DT does not clear our training-task attainment line, so it was demoted to a case study before any of its out-of-hull numbers were read and it never appears in a performance comparison with Prompt-DT. Seeds beyond its third were never trained, so every number here is over three seeds while the grid is over six. Its attainment is 0.487, 0.350 and 0.551 at the end of training, 0.462, 0.449 and 0.322 averaged over the last five evaluation rounds, and 0.734, 0.704 and 0.615 at its best round, against a line of 0.80 and a three-seed Prompt-DT reference of 0.898 ± 0.009; on cold-start rollouts it reads 0.150, 0.386 and 0.274, and no task out of twenty clears the line on any seed. Its band returns are -61.16 ± 5.44, -196.55 ± 33.55 and -339.81 ± 22.76 few-shot and -71.33 ± 2.70, -231.11 ± 5.88 and -367.44 ± 9.04 zero-shot, against experts of -22.06, -25.66 and -31.92.

What makes the agent worth a case study is a split between its channels, which Table 28 reads one channel at a time.

Table 28: Meta-DT on HalfCheetah-Vel, three seeds, few-shot, this agent having no further seeds. Bold marks the largest ratio of the above-hull slope to the in-hull slope; it does not establish hull exit. Below-hull slopes are omitted throughout, and the Prompt-DT reference row is the six-seed grid value.

Channel (few-shot)	in hull	above hull	above / in hull
code probe, shipped encoder	$+0.556 \pm 0.078$	$\mathbf{+0.363 \pm 0.248}$	$\mathbf{+0.65}$
code probe, encoder at epoch 100	$+0.916 \pm 0.009$	-0.166 ± 0.065	-0.18
residual-stream probe	$+0.914 \pm 0.008$	$+0.205 \pm 0.089$	$+0.22$
layer-0 control	$+0.855 \pm 0.006$	$+0.224 \pm 0.101$	$+0.26$
soft action inversion	$+0.844 \pm 0.152$	$+0.100 \pm 0.017$	$+0.12$
hard action inversion	$+1.056 \pm 0.014$	$+0.228 \pm 0.057$	$+0.22$
achieved speed	$+1.070 \pm 0.047$	$+0.197 \pm 0.057$	$+0.18$
Prompt-DT reference, soft inversion	$+1.008 \pm 0.004$	$+0.151 \pm 0.031$	$+0.15$

Below-hull slopes are left out of that table because on this agent that side is a cliff on which one or two tasks near the boundary survive, which a linear slope would misdescribe. The behavioral readout changes little outside the hull, as in Prompt-DT, while the shipped encoder's explicit-code probe has a larger ratio. This pattern is consistent with attenuation after the encoder, but it does not localize a causal mechanism since the readout is fitted and the channels were not intervened on independently. However, the surviving code signal depends on when the encoder stopped: recomputing the code on the same rollouts with the epoch-100 encoder collapses the ratio from $+0.65$ to -0.18. The collapse is in step with the encoder's own held-out decoding loss, which rises from 0.0054 at the best epoch to 0.090 on the first seed and from 0.0037 to 0.117 on the second. The right reading is therefore not that the encoder extrapolates but that an early-stopped encoder has not yet destroyed a signal present in its input, and under zero-shot evaluation the logged code's above-hull slope is -0.106 ± 0.026 and the effect is gone. The trade-off has to be stated with it: the shipped encoder's code has a weak in-hull gain to begin with, slope 0.556 with training-task cross-validated R^2 of 0.6440, 0.4868 and 0.4567, all under the line, while the epoch-100 code clears the line at 0.8956, 0.9100 and 0.9250 and dies out of hull. Meta-DT's training-task inversion rate is 0.1756, 0.3018 and 0.2804 against 0.91 for Prompt-DT, so its inversion rows are supporting evidence only, and we report no absolute calibrated code values for this agent, since the ridge readout is fitted on the training distribution and extrapolates arbitrarily where the reward magnitude leaves it.

H Reproducibility

This appendix settles what a reader can check, and how a number in a table can be traced back to the run that produced it. We will say what the release contains, then what makes a table cell checkable not merely plausible.

H.1 What is released

The release carries the whole path from the environment to the printed number. It contains the environment and family definitions together with the manifests that fix every task parameter and every split, the conversion pipeline that turns raw expert rollouts into the per-task trajectory files of Appendix B, and the trainers for every arm in this paper. It also contains the rollout logger that records states, actions and hidden states, the three instruments, the per-family aggregators that turn logged rollouts into slopes and band means, the merge step that assembles the grid and the manipulation tables, and the one script that draws every figure and writes every generated table here. The aggregate outputs those tables are read from ship with it, so a reader can check a number without a GPU and without rerunning anything. Checkpoints are organized by family, arm and seed, and each run directory keeps the exact configuration it was launched with together with a newline-delimited training log, so a hyperparameter quoted in Appendix C can be checked against the run that produced the number rather than against our description of it.

H.2 How a number can be traced

Every measurement in this paper is identified by five keys, the family, the architecture, the arm, the protocol and the band, and those five keys are also how the released analysis tree is arranged, so a cell is found by opening the aggregate output for its family and arm and reading off the band. Two properties of the pipeline are what make such a cell checkable. Slopes and band means are computed once, by the per-family aggregators, and the merge and table steps never refit them, so a table cell is a transcription of an aggregate output not a second computation that might disagree with the first. And the figure script asserts its own inputs before it draws anything: it reproduces the two HalfCheetah-Vel saturation cells of Table 1, $+1.1280 \pm 0.0590$ and $+2.9860 \pm 0.0230$, and, on the first three seeds, the entire control column of the widening band file to 10^{-6}, and it stops instead of drawing if either check fails. Note that these assertions run on every build of the paper, which is what keeps a figure and the table beside it from drifting apart.