WHAT A RELATIVE IMPROVEMENT MEASURES: REFERENCE VALUES FOR LATENT-REGIME REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Work on reinforcement learning under non-stationarity reports adaptation as a percentage improvement over a non-adaptive baseline and compares algorithms using that number. This approach merges the adaptation the environment offers, the fraction the agent collects, and how far the baseline falls short. Since only the fraction belongs to the algorithm, the same improvement means different things in different environments. The three enter the reported improvement through an exact identity, so any reported improvement already bounds the adaptation that an algorithm collected. Measuring the fraction instead needs a policy optimal under the agent's own information, which commonly used environments do not carry. In this work, we define a class in which that policy, together with a controller told the regime and one whose belief never moves, follows from the environment's parameters, and release 21 of its configurations. We train seven algorithms on them, and across runs the reported improvement varies by a factor of 17 while the fraction collected varies by a factor of 1.2, so almost everything separating the results is the environment. That term is set before any agent trains, and on regime processes fitted to daily equity returns, selling down a stock position and hedging an option differ tenfold in what adaptation is worth. We therefore report the fraction collected together with the reported improvement. Our configurations give that fraction exactly, and outside the class a published number still bounds it, while one algorithm trained twice, once with the regime revealed and once without, estimates from the gap between the two costs what the regime label itself is worth there.

1 INTRODUCTION

Reinforcement learning under non-stationarity is usually studied from the algorithm's side (Padakandla, 2021; Khetarpal et al., 2022). An environment changes, an agent adapts, and the adaptation is reported as its relative improvement over a baseline that does not adapt, a number we denote $\mathcal{I}$, which appears variously as a normalized score, a normalized return, or forward transfer. However, such a number is an estimate of three quantities at once.

We denote by $\mathcal{A}$ the value of adaptation that the environment offers. It is the cost the best policy removes by inferring the regime from the observations, against a policy whose belief never moves and therefore cannot react when the regime changes. We write every quantity as a cost, so smaller is better, and a benchmark stated in returns enters as the distance from the best return the problem allows. We denote by s the fraction of that adaptation an agent collects, and by ε the fraction by which the non-adapting baseline's cost exceeds that of the best policy that also never adapts. The three determine the relative improvement:

$$\mathcal{I} \;=\; \frac{\mathcal{A}\, s + \varepsilon}{1 + \varepsilon}.$$

The identity follows in one step from the three definitions, which we write out in costs in Section 3. When the baseline reaches its own optimum, ε is zero and $\mathcal{I}$ is exactly $\mathcal{A}\, s$, and otherwise $\mathcal{I}$ is larger. One number cannot identify three, and of the three only s belongs to the algorithm, which is the quantity a comparison across environments should rest on.

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.

Computing s needs $\mathcal{A}$, and computing $\mathcal{A}$ needs a policy that is optimal under the agent's own information set. The environments in common use for non-stationarity in deep reinforcement learning carry no such optimal policy, and Tao et al. (2025) take that reference from trained agents in nearly every environment of POBAX, the closest benchmark to ours. In Section 6, we run this decomposition on its suite, where one of those trained references falls below an optimum the environment itself fixes.

We make $\mathcal{A}$ computable and still leave the agent a representation-learning problem. Our environment is a partially observable Markov decision process (POMDP) whose hidden part is a two-state Markov chain, the regime, and an episode runs 26 steps, long enough for the chain to switch about once at the persistences we use. The state and the action are each a real number, and the stage cost charges quadratically for how far the agent moves the state and for how far it ends from a target, so within a known regime the problem is linear-quadratic tracking, whose optimum follows from a Riccati recursion (Chizeck et al., 1986). The regime moves the two cost weights, the target, and the observation the agent sees, and we vary them one at a time and in combination. The agent does not observe that structure, and its input is 74-dimensional, a window of past observations together with a fixed random embedding of the state and the time remaining, so the policy has to decode a representation rather than read off the optimum's closed-form action. We train seven model-free algorithms on it, six standard and one we build for the problem.

We compute three controllers from the model parameters, our reference points, and place every agent between their costs, the reference values, which is how we measure s. The regime-informed controller is told the regime before each decision. The belief-optimal controller carries the posterior over the regime implied by its observations, so no agent restricted to that history does better. Instead, the uninformed controller's belief never moves. $\mathcal{A}$ is the distance from the uninformed controller to the belief-optimal one, the term missing from $\mathcal{I}$, and the value of privileged regime access $\mathcal{V}$ is the distance from the belief-optimal controller to the regime-informed one, which no adaptive agent reaches. Both are normalized values of information (Howard, 1966; De Lara & Gossner, 2020), $\mathcal{V}$ of the regime label and $\mathcal{A}$ of the observation channel, and $\mathcal{V}$'s unnormalized gap is the standard bound of partially observable planning (Hauskrecht, 2000). Where the regime is easy to infer from the observations, filtering already reaches nearly the regime-informed cost, so $\mathcal{V}$ is near zero while $\mathcal{A}$ is at its largest. Therefore, scoring agents against a policy given the true regime reports $\mathcal{V}$ rather than $\mathcal{A}$, and returns near zero on the environments where adaptation is worth the most.

Our contribution is to separate the three terms of $\mathcal{I}$, and show what is required to obtain them. The decomposition and the bound that follows from it need only the reported percentage, so any published improvement gives a ceiling on the adaptation that the algorithm collected, without access to the environment and without a new training run. We apply them to Continual World (Wołczyk et al., 2021) from the numbers it reports, and to POBAX (Tao et al., 2025), whose reference we rerun to check. Neither carries a policy optimal under its agents' information, so the ceiling is all their published results can give, and separating the terms needs an environment whose reference values are known in advance. We define a class of environments whose three reference values follow from the model itself, and release 21 of its configurations as a suite, on which s is measured rather than bounded. Estimating $\mathcal{V}$ requires no model at all, since we train one algorithm twice on the same environment, once with the regime revealed and once without, and the gap between the two costs estimates what the regime label is worth there. In Section 3, we derive the decomposition and define the class, in Section 4 we read what a configuration offers from its parameters before training on it, in Section 5 we train the seven algorithms against the reference points, and in Section 6 we measure all three terms. In Section 7, we fit the regime process to daily equity returns.

2 Related Work

Theoretical work on non-stationary reinforcement learning typically measures an agent against the per-episode optimal policy by dynamic regret under a variation budget, and the bounds come from sliding windows, restarts, change detection and function approximation (Cheung et al., 2020; Mao et al., 2021; Ding et al., 2023; Feng et al., 2023; Li et al., 2024). Lee et al. (2025) analyze multi-task representation learning for linear-quadratic adaptive control, whose linear-quadratic control problem our class is also built on, with the unknown in the system dynamics rather than in the cost. These

bounds are on the gap between an agent and the per-episode optimum, as a rate in the horizon, while $\mathcal{A}$ measures the gap from a policy that never adapts to the best one restricted to the observations.

Our environment has an unobserved parameter that follows a Markov chain and moves the stage cost, which places it among latent-regime and hidden-mode formulations (Zhou et al., 2021; Zhang et al., 2021; Feng et al., 2022; Poiani et al., 2021). Lecarpentier et al. (2021) bound how much the optimal value can move when two such tasks are close in a metric over rewards and transitions, and Agarwal et al. (2021a) build a policy similarity metric between states from their optimal policies. The Lipschitz bound rests on a distance between the two tasks' own parameters. We find that $\mathcal{A}$ and $\mathcal{V}$ depend on how sharply an observation reveals the regime and on the distance between the two regimes' optimal actions, both computed from the environment's model parameters.

Among quantities computed from a trained agent, the closest to ours is the λ-discrepancy of Allen et al. (2024), shown to be exactly zero for every Markov decision process (MDP) and almost always non-zero for a broad class of partially observable environments. It reads the agent's value estimates to detect whether state information is missing, while we compute what that information is worth from the model parameters before any agent is trained.

Among benchmarks that fix a reference point, Tao et al. (2025) call an environment *memory improvable* when agents with more and with less state information perform differently, and run one algorithm at both levels to get a *floor* and a *ceiling*. That gap both decides which environments qualify for POBAX and bounds every agent it evaluates. Both endpoints are trained agents in nearly every domain, so the gap is a property of the agent as much as of the environment. Deb & Krishnan (2026) score large language model agents between a floor that ignores the symptoms it observes and an exact Bayes filter, and Ibrahim et al. (2026) generate systems whose optimal policy is known by construction. Exact optima are otherwise standard only where the environments are small enough to solve (Osband et al., 2020; Rajan et al., 2023). Wołczyk et al. (2021) note that the forward transfer measure of Continual World is ill-defined where the baseline already reaches its ceiling, so that work avoids those tasks while we decompose the quantity. None of these benchmarks carries an optimum under two stated information sets, which our class carries in every configuration.

The belief-optimal controller is the analogue, for a moving regime, of the Bayes-optimal policy of a Bayes-adaptive MDP (Duff, 2002), whose unknown parameter is fixed, and meta-reinforcement learning approximates that policy. Zintgraf et al. (2020) learn an approximate belief over the unobserved task, Rakelly et al. (2019) infer a probabilistic context from recent experience, and Duan et al. (2016) leave the inference implicit in a recurrent policy. Each approximates the belief-optimal controller, which we compute numerically and use to measure what the environment offers rather than to act in it.

In asymmetric reinforcement learning, Ebi et al. (2026) ask which privileged signals are worth supplying to a critic, and answer with two informativeness criteria, one applied before training and one after. We instead measure how much a hidden regime is worth to any agent. Brini & Kolm (2026) benchmark PPO on a simulated trading problem with a closed-form optimal policy and Cui et al. (2026) prove convergence of value iteration for stochastic linear-quadratic control learned from data alone, but neither carries a latent regime.

Evaluation methodology has concentrated on sampling uncertainty, on aggregate point estimates that ignore what a handful of runs implies (Agarwal et al., 2021b), on metrics that make reported results inconsistent (Jordan et al., 2020), and on the assumptions those measures rest on (Patterson et al., 2024). Each of these corrects the estimator of a reported number, while Equation (4) says the number is a combination of three quantities, so more seeds estimate that combination more accurately without separating its terms. Henderson et al. (2018) illustrate the variability in reported metrics and results when algorithms are compared against common baselines, and Ni et al. (2022) find that a carefully implemented recurrent baseline matches or beats six specialized methods on 18 of 21 environments. Both document how far a baseline falls short, which is ε, one algorithm at a time. Gruslys et al. (2018) note that the ten "most score-generous" Atari games can explain more than half of the variance across algorithms in mean human-normalized score, and there the environment's term dominates the table. Mesbahi et al. (2025) argue that tuning over an agent's whole lifetime, the full stream of tasks it meets, invalidates continual comparisons, and we fix every threshold before the run it applies to.

3 SETTING AND THE THREE REFERENCE POINTS

We study finite-horizon problems whose source of non-stationarity is a latent regime $z_t \in \{1, \dots, N\}$ that follows a Markov chain with transition matrix P and that nothing the agent does influences. We write it for general N and report two regimes throughout. The agent never observes z_t and sees instead a scalar regime signal $o_t \sim \mathcal{N}(\mu_{z_t}, \sigma^2_{z_t})$ whose mean and variance depend on the current regime. The observation can be any regime-dependent scalar, and in the instances we calibrate to data in Section 7 it is a daily log financial return.

Within a regime the agent faces a linear-quadratic tracking problem in a scalar state x_t, which it controls and which its policy sees through a fixed random embedding, separately from o_t (Appendix B.1). It takes an action u_t that sets where the state goes next, and pays

$$c_{z_t}(u_t - x_t)^2 + \rho_{z_t}(u_t - m_{z_t})^2, \qquad x_{t+1} = a\,u_t + d, \tag{1}$$

over $t = 0, \dots, T-2$, with the gain a and the drift d fixed across regimes. The regime sets three numbers in this cost: the cost of acting c_z prices moving the state, the tracking weight ρ_z prices ending away from the target, and the target m_z is where the regime wants the state to be. We require the last action to return the state to zero and pay only the cost of acting, so an episode of T steps leaves $T-1$ free decisions. The stage cost is the negative reward and z_t is the hidden part of the state, so this is a POMDP whose latent state follows a Markov chain. The regime enters the cost and not the dynamics, which makes this a special case of a Markov jump linear system with a hidden mode (Ogura et al., 2018). In Section 4, we recompute the same quantities outside this class. We call (c_z, ρ_z, m_z) the *control parameters*, which set how far apart the regimes' optimal actions are, and (μ_z, σ_z) the *observation parameters*, which set how identifiable the regime is, that is, how well an agent can tell from the regime signal which regime it is in.

Three controllers follow from the model parameters before any agent exists. They are the *reference points*, and the costs they attain are the *reference values*. The *regime-informed controller* is optimal when the privileged signal z_t is revealed before each decision. The cost it pays from any state onward is quadratic in x for every regime and horizon, so backward induction over the N regimes computes it in closed form by a Riccati recursion, which we write out in Appendix A. No agent restricted to the observations improves on its cost J^{inf}, with equality where the regime is fully identified or does not change the optimal action. Every J here is an expected cost.

The *belief-optimal controller* is the best an agent restricted to the observations can do. It carries the filtered posterior $b_t = \Pr(z_t \mid o_1, \dots, o_t)$, updated by the Bayes filter for a hidden Markov chain, known in econometrics as the Hamilton filter (Hamilton, 1989). We use the filtered posterior, since a smoothed one would condition on observations that no agent has at decision time. The belief update does not depend on the action, so the problem reduces to a fully observed MDP over (x_t, b_t) (Åström, 1965), again quadratic in x at every belief. This controller is the one of the three we solve numerically, by backward induction over a discretized belief space (Appendix B.3), and its cost is J^{bel}.

The *uninformed controller* is optimal for an agent whose observations carry nothing about the regime. We draw z_0 from the chain's stationary distribution, which a two-state chain with persistences strictly inside $(0, 1)$ has and which is unique, so this controller's belief is correct and constant throughout, and its cost J^{noinf} is the *no-information bound*.

We measure the gap between the belief-optimal and the regime-informed controllers, as a fraction of the achievable cost J^{bel},

$$\mathcal{V} \;=\; \frac{J^{\mathrm{bel}} - J^{\mathrm{inf}}}{J^{\mathrm{bel}}}, \tag{2}$$

which we call the *value of privileged regime access*. No policy measurable with respect to the state and the observation history improves on J^{bel}, since the belief is a sufficient statistic for that history, meaning that conditioning on b_t leaves the same distribution over z_t as conditioning on the whole history. $\mathcal{V}$ is therefore the part of the problem no adaptation mechanism reaches, and in Section 5 we test whether it can be measured from trained runs. Adaptation earns the distance an agent covers by updating its belief at all rather than holding it fixed, from the uninformed controller down to the belief-optimal controller, and we call that distance the *value of adaptation*, a fraction of the no-information bound,

$$\mathcal{A} \;=\; \frac{J^{\mathrm{noinf}} - J^{\mathrm{bel}}}{J^{\mathrm{noinf}}}. \tag{3}$$

Every environment carries both numbers, against different denominators, and the ordering $J^{\mathrm{inf}} \leq J^{\mathrm{bel}} \leq J^{\mathrm{noinf}}$ puts both in $[0, 1)$.

As an example, we take two regimes which share both cost weights, $c = (2, 2)$ and $\rho = (1, 1)$ with one entry per regime, and differ only in their targets, $m = (0.3, 0.7)$, over an episode of $T = 26$ steps from $x_0 = 0$. They persist at self-transition probabilities 0.97 and 0.92, and each step produces an observation drawn from $\mathcal{N}(\pm 0.05, 0.02^2)$ according to the current regime. The regime-informed controller pays 0.53, the belief-optimal controller 0.66, and the uninformed controller 1.13. Adaptation is worth $\mathcal{A} = 42\%$ here and privileged access $\mathcal{V} = 20\%$, and the two compose to a total of 53% rather than to their sum, since each is taken against a different cost, as we show in Equation (6).

A non-adapting baseline is an agent trained without the regime signal o_t, so it still sees the state x_t but cannot infer the regime, and the relative improvement reported for an adaptive agent is the fraction by which it improves on that baseline's evaluation cost J^{base}. The three costs above come from controllers we compute from the model, while J^{base} and the agent's own evaluation cost J^{agent} are measured from trained runs, which is why a baseline can cost more than the controller it stands for. We denote by $s = (J^{\mathrm{noinf}} - J^{\mathrm{agent}})/(J^{\mathrm{noinf}} - J^{\mathrm{bel}})$ the fraction of the available adaptation the agent collects, which is undefined when $\mathcal{A}$ is zero, and by $\varepsilon = J^{\mathrm{base}}/J^{\mathrm{noinf}} - 1$ the fraction by which the trained baseline's cost exceeds J^{noinf}, the cost of the uninformed controller it stands for.

Proposition 3.1. *For an agent's relative improvement $\mathcal{I}$ over a non-adapting baseline,*

$$\mathcal{I} \;=\; \frac{J^{\mathrm{base}} - J^{\mathrm{agent}}}{J^{\mathrm{base}}} \;=\; \frac{\mathcal{A}\, s + \varepsilon}{1 + \varepsilon}, \tag{4}$$

and, since a baseline cannot improve on the policy it stands for and costs are positive,

$$\mathcal{A}\, s \;=\; \mathcal{I} - \varepsilon\,(1 - \mathcal{I}) \;\leq\; \mathcal{I}, \tag{5}$$

with equality only at $\varepsilon = 0$.

Both follow by substitution. By the definitions of $\mathcal{A}$ and s the agent pays $J^{\mathrm{agent}} = J^{\mathrm{noinf}}(1 - \mathcal{A}\, s)$ and the baseline pays $J^{\mathrm{base}} = J^{\mathrm{noinf}}(1 + \varepsilon)$, which gives Equation (4) and rearranges to Equation (5). No policy beats the optimum of its own class, so $\varepsilon \geq 0$, and positive costs put $\mathcal{I}$ below one. The identity holds run by run, and Equation (5) holds for any reported relative improvement, whether or not $\mathcal{A}$ can be computed, so $\mathcal{I}$ overstates what the algorithm itself contributed. If we assume a ceiling $\bar{\varepsilon}$ on ε, then $\mathcal{A}\, s$ lies between $\mathcal{I} - \bar{\varepsilon}(1 - \mathcal{I})$ and $\mathcal{I}$, which needs no value for $\mathcal{A}$ and no further run. Two algorithms compared on the same environment against the same baseline share both $\mathcal{A}$ and ε, so $\mathcal{I}$ ranks them correctly there but cannot compare improvements reported on different environments, which carry different $\mathcal{A}$ and ε.

Remark 3.1. We fix the control problem and the transition matrix P, and vary only the observation parameters (μ_z, σ_z). The composed total $1 - J^{\mathrm{inf}}/J^{\mathrm{noinf}}$ is then invariant, and $\mathcal{V}$ and $\mathcal{A}$ compose multiplicatively into it through

$$(1 - \mathcal{V})(1 - \mathcal{A}) \;=\; \frac{J^{\mathrm{inf}}}{J^{\mathrm{noinf}}}, \tag{6}$$

with $\mathcal{V}$ increasing and $\mathcal{A}$ decreasing in the cost J^{bel} the belief-optimal controller pays.

We derive it in Appendix A. Identifiability decides only how the composed total splits between them, so a benchmark that reports $\mathcal{V}$ alone cannot say what adaptation was worth.

4 Where Regime Information Has Value

By Proposition 3.1, $\mathcal{A}$ is the one term $\mathcal{I}$ cannot give, and in Remark 3.1 we tie it to $\mathcal{V}$. Existing surveys sort non-stationary problems by what the shift touches (Run & Ding, 2026; Wibowo et al., 2026), and we build those categories inside our class, splitting the cost category in two, so that each one carries a computed value instead of a qualitative label. In the *cost-weight* component the regime moves the cost of acting c_z and the tracking weight ρ_z while both regimes share one target. In the *cost-target* component it moves the target m_z while c_z and ρ_z are common. In the *observation* component the control parameters do not move, so the two regimes' observation distributions overlap more or less while what to do in each stays fixed. We first vary one component at a time over 134 configurations,

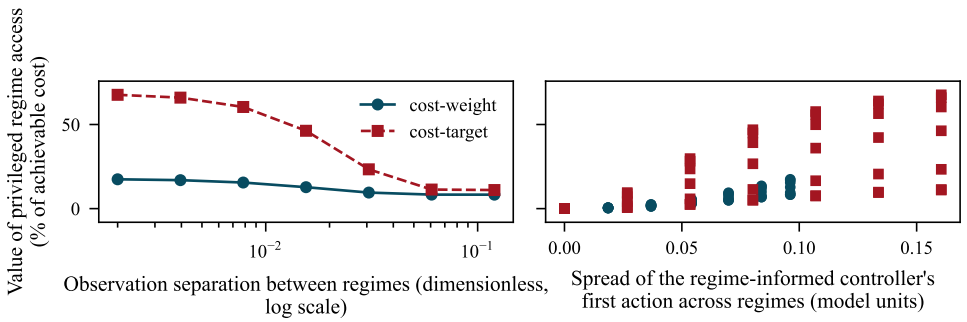


Figure 1: Seven levels of each of the cost-weight and cost-target components' control parameters, paired with seven observation separations, for 98 configurations. Left, the largest value of privileged regime access $\mathcal{V}$ that each component reaches at each observation separation, the offset of each regime's observation mean from zero. Right, the same $\mathcal{V}$ against the displacement diagnostic, the spread of the regime-informed controller's first action across regimes. The grid and the regression behind the right panel are in Appendix B.2.

each one a setting of the parameters and so an environment of its own, and the largest $\mathcal{V}$ a component reaches runs from 9.4% for cost-weight to 65.4% for observation (Appendix Figure A5).

In that grid we hold the *observation separation* fixed across the two cost components, the offset of each regime's observation mean from zero, which sets how identifiable the regime is at a fixed observation standard deviation. $\mathcal{V}$ moves most along that axis, so which of the two cost components carries the larger $\mathcal{V}$ can change as the separation changes, and we cannot rank them from a grid that holds it fixed. We therefore cross each cost component's control parameter with the observation separation, the cost of acting c_z for cost-weight and the target m_z for cost-target, holding the tracking weight ρ_z common in both (Figure 1). Over the range of values we search, which sets each component's scale, cost-target exceeds cost-weight at every observation separation, by a factor of 3.88 where the regime is hardest to identify and 1.33 where it is easiest.

The spread of the regime-informed controller's first action across regimes, the *displacement diagnostic*, and the logarithm of the observation separation together account for 64.8% of the variation in $\mathcal{V}$ across the crossed grid. $\mathcal{V}$ is largest where the regime is hard to identify and $\mathcal{A}$ where it is easy. On average over the control settings, $\mathcal{V}$ and $\mathcal{A}$ are 23.2% against 0.6% at the smallest observation separation we search and 4.3% against 20.7% at the largest.

The three reference values need computable controllers rather than closed forms, so we ask whether the same trade-off holds outside the linear-quadratic class, where we solve all three by value iteration instead. An agent moves along a chain of five positions with three actions that sometimes fail and is rewarded for standing on a goal that depends on an exogenous hidden mode. We cross 84 configurations over how far apart the two modes put that goal, how long a mode persists, and how much the signal reveals about it (Appendix Figure A6). Averaged over the configurations where the two modes' goals differ, $\mathcal{V}$ falls from 26.5% at a signal that carries nothing about the mode to zero at one that reveals it, while $\mathcal{A}$ rises from zero to 26.5% over the same range, and the composed total holds fixed to the last digit we record, which checks the recomputation rather than the algebra. The displacement diagnostic transfers in sign, since it and $\mathcal{V}$ are both zero where the goals coincide, while its ordering weakens to a rank correlation of 0.485 over the 54 configurations where $\mathcal{V}$ is above zero.

5 Do Agents Reach Their Own Reference Point?

Inside the linear-quadratic class of Section 3, we measure what an agent recovers against what its own information makes achievable. We take the two-run construction of Tao et al. (2025) and run it on every algorithm twice per configuration, once in the uninformed condition and once in the informed condition, which appends the true regime as a one-hot. We compare the uninformed run to the belief-optimal controller, the informed run to the regime-informed controller, and the measured gap between the two runs to $\mathcal{V}$, a comparison that needs no model. A third condition, masked, zeroes

the observation window and stands in for the non-adapting baseline of Section 6. The uninformed controller bounds the masked condition, while the belief-optimal controller bounds the uninformed condition, which still reads the window. We train seven algorithms, six of them standard, feedforward and recurrent Proximal Policy Optimization (PPO) (Schulman et al., 2017), Soft Actor-Critic (SAC) (Haarnoja et al., 2018), Twin Delayed Deep Deterministic policy gradient (TD3) (Fujimoto et al., 2018), RL^2 (Duan et al., 2016) and VariBAD (Zintgraf et al., 2020). The seventh we build for this problem, a context encoder that compresses the observation window into an 8-dimensional vector the policy conditions on. We give RL^2 and VariBAD the observation stream and the previous action, which is the information set the belief-optimal controller bounds, at the budget their own calibration selected (Appendix H). We train all seven at six seeds per configuration and condition, eight in the identifiability and inventory ranking studies, and we rerun at 30 seeds any verdict, meaning any outcome of a threshold fixed in advance, whose bootstrap interval still covers that threshold. We evaluate every run over 8,192 episodes, so every run carries the same evaluation sample size, and every interval is a percentile bootstrap over seeds (Appendix B.4).

We run three studies on constructed configurations, each with its hypotheses and thresholds written down before the corresponding runs, so that no threshold could be chosen after seeing the result (Table 1). In the inventory ranking and target-separation studies, we test whether the measured gap ranks $\mathcal{V}$. We run the first on the inventory problem, an instance of the class in which the agent unwinds a position while its own trading moves the price against it (Section 7). We run the second on the cost-target component as a scalar tracking problem over five target separations, the distance between the two regimes' targets m_z. At zero separation the two regimes are identical and $\mathcal{V}$ is zero, which is the control the design carries for free, and widening the separation pulls the two regimes' optimal actions apart and raises $\mathcal{V}$ to 34.5% (Appendix Figure A4). In the identifiability study, we test whether an agent's shortfall grows as the regime becomes harder to identify, on the inventory-style control problem, a tracking problem whose cost of acting differs across regimes (Appendix B.2), across observation separations. In a fourth study, the memory study, we vary how much of the observation history the policy reads, with no hypothesis fixed and no verdict (Appendix M), and we report a fifth on calibrated configurations in Section 8. A sixth varies the length of the value backup inside PPO, against a prediction we fixed in advance about where bootstrapping should cost an agent (Appendix E).

We report an algorithm's result only when its informed run comes within Table 1's tolerance of the regime-informed controller. An algorithm that misses this validity check has not solved the problem even with the regime revealed, so its uninformed run says nothing about what adaptation was worth. On the cost-target component four of the seven pass, at worst 10.8% above their own controller, averaged over 30 seeds. TD3's cost diverges by orders of magnitude, because its deterministic actor assumes a bounded action space while the action here is unbounded. We keep it unbounded because the controllers minimize over the whole real line, and because bounding it trades TD3's failure for SAC's and recurrent PPO's. Of the two meta-reinforcement-learning algorithms, VariBAD passes and RL^2 is unstable, with an informed run 98.5% above its controller at the smallest non-zero separation, so we exclude RL^2 here, as the same validity check excludes SAC on the inventory problem. With the previous reward as well, as in the original implementations, the input of RL^2 and VariBAD exceeds what the belief-optimal controller conditions on, so that controller no longer bounds them, and we report those runs in Appendix H. The context encoder misses the same check at six seeds, at 66.0% above its controller, with a gap of 4.3% where $\mathcal{V}$ is zero.

The four that pass recover $\mathcal{V}$ from their two trained conditions across the whole range, at a rank correlation of 1.000 for each, so the two-run estimate orders the configurations the way $\mathcal{V}$ does. The mean absolute error between the measured gap and $\mathcal{V}$ across the five separations runs from 0.4 percentage points for SAC to 3.0 for recurrent PPO. The worst single configuration is recurrent PPO at the smallest non-zero separation, where the measured gap is 13.0% against a $\mathcal{V}$ of 6.5%, the separation at which its own informed run is weakest. The gap measured with feedforward PPO ranks $\mathcal{V}$ on the inventory problem too, over 12 configurations (Appendix F). The identifiability study's hypothesis comes back unsupported, since an agent's shortfall does not grow where $\mathcal{V}$ does (Appendix G).

We release as an installable environment suite the 21 constructed configurations we run the inventory ranking, target-separation and memory studies on, each registered by name and supplied in the three conditions with its three reference values (Appendix B.2). We train one algorithm in both conditions

Table 1: Verdicts on the hypotheses and predictions we fixed before the corresponding runs. r is the Spearman rank correlation. In the validity-check row, we pool both problems. In the zero-value control row, we include only the four algorithms that pass that check on the cost-target component. Bold marks the checks our own runs failed, as distinct from unsupported hypotheses. *Rejected* means a claim stated against the instrument did not hold, and *unsupported* and *refuted* mean a hypothesis or a prediction of ours did not.

Study	Claim tested	Threshold	Outcome	App.
Target separation	gap misranks $\mathcal{V}$	$r < 0.7$	rejected, $r = 1.000$	H
Inventory ranking, primary	gap misranks $\mathcal{V}$	$r < 0.7$	rejected, $r = 0.993$	F
Inventory ranking, secondary	inverted pair, disjoint intervals	≥ 1 pair	none of 66	F
Identifiability, primary	uninformed shortfall grows	$r > 0.7$	unsupported, -0.257	G
Bootstrapping prediction	low bootstrapping hurts uninformed only	informed penalty ≈ 0	refuted, informed 0.63%	E
Zero-value controls	measured gap at $\mathcal{V} = 0$	$\lvert\text{gap}\rvert \leq 1\%$	**1 failed**, 3 passed	H
Informed-run validity check	informed run reaches its controller	within 15%	**4 failed**, excluded	B.4
Identifiability, secondary	informed shortfall flat	$\lvert r\rvert < 0.5$	**failed**, range 0.05%	G
Calibrated agents	agents reach their controllers	within 15%	**all failed**, not read	J

to separate a control failure from an inference failure, which excludes TD3 and SAC here and tells an agent trained anywhere else that a missed informed run is not a memory problem (Appendix L).

6 What the Relative Improvement Measures

We measure all three terms of Proposition 3.1 together on the runs of Section 5, over 28 agent-environment pairs, drawn from the cost-target component and from the memory study's inventory-style control problem (Appendix B.2). The masked condition supplies the baseline cost J^{base}, so ε is the fraction by which it exceeds the no-information bound, and the reference values supply $\mathcal{A}$ and s.

The reported number and the fraction the algorithm captured do not order these runs the same way. Seven of the 28 pairs are set aside, four for an undertrained baseline and three where the environment offered under 1%. Across the remaining 21, $\mathcal{I}$ ranges from 3.2% to 55.8% while s runs from 83.0% to 102.1%, and since no policy can beat its own bound the single reading above 100% is noise. So $\mathcal{I}$ varies by a factor of 17 across these runs while s varies by a factor of 1.2, and ranking by the two disagrees on 8 of the 63 pairs where the comparison is admissible, meaning both orderings resolve, the two runs share a problem, neither baseline is undertrained, and neither algorithm came in under its own bound (Appendix I and Appendix Figure A7). Feedforward PPO on the inventory-style control problem collects 97.3% of what was available and reports $\mathcal{I} = 6.1\%$, while recurrent PPO on the cost-target component collects 92.1% and reports 40.4%.

A reported number can be read at face value only where the baseline reaches its own bound and ε is near zero. Where the baseline is undertrained, ε reaches 54% and $\mathcal{I}$ reads 38.8% to 46.8% while only 19.3% was available, so the reported number is more than twice what the environment could give.

We apply the bound to two published suites, the first with no new runs at all. Wołczyk et al. (2021) report forward transfer on Continual World as the area under an algorithm's success curve on a task, above the area that training the task from scratch reaches, divided by the room the from-scratch run leaves. That measure is Equation (4) with one minus the area as the cost and the from-scratch run as the baseline. Continual World's best algorithm by average performance, PackNet (Mallya & Lazebnik, 2018), reports 0.19 with a 90% interval of $[0.15, 0.23]$ over 20 seeds, against a reference for what the sequence offers of 0.46. Composing several tasks can exceed that, so $\mathcal{A} \geq 0.46$, and Equation (5) therefore caps PackNet at 41% of what was available. All its algorithms share one reference, so $\mathcal{I}$ ranks them as s would.

With the second suite, we check the reference point itself. We run the two-run construction on POBAX (Tao et al., 2025), whose screen supplies all three information sets at hyperparameters it selected, observations without memory, the full state without memory, and the memory-equipped runs. Over four domains at 30 seeds in every cell that runs, recurrent PPO collects 87.9% of what was available on one and -0.019% on another, while reporting 131.2% and -0.11%. At equal weights the ordering of its three algorithms survives, but the reference does not, since it is itself a trained run, and on the T-maze it reaches 3.939 where the environment pays exactly 4.0, which puts an algorithm at 102.6% of what was available (Appendix K). On POBAX, where an optimum is fixed by the

environment rather than by a run, the reference point is what fails, which is the failure a computed reference removes.

7 Calibration to Real Data

Every configuration so far is one we chose, so the quantities could be large because we made them so. Here we fit the environment's parameters to daily US equity returns instead, and read the reference values on two financial control problems that embed in the class of Section 3, unwinding an inventory of a financial asset and hedging a financial option, where a latent regime in volatility, how far returns swing day to day, is the non-stationarity a practitioner actually faces. Those returns enter only through the fitted parameters, and no agent trains on a real price series, since the belief-optimal controller filters against a known generative model and a single historical price path does not come with one. Liquidating an inventory (Almgren & Chriss, 2001) falls in the cost-weight component. The state is the position left to sell, each trade moves the price in proportion to its own size under the linear-impact model, the target is zero in every regime, and a regime that raises volatility and trading cost moves only what acting and waiting cost. In hedging a European option the agent holds a position in the underlying stock, pays to change it, and pays for any distance between it and the option's delta, which is the stock holding that offsets the option's exposure. The regime moves both costs and that target, since a regime that moves volatility moves the delta (Black & Scholes, 1973; Naik, 1993).

We fit a two-state Markov-switching model with a regime-dependent mean and variance to the daily returns of 500 US stocks from the Center for Research in Security Prices (CRSP) over 2000 to 2024, drawn 50 per decile of daily dollar volume among those with at least fourteen years of data, which spreads the sample across liquidity because dollar volume enters the price-impact coefficient directly. Of these, 496 pass a screen we fixed before fitting anything (Appendix C). We fit one regime process per stock, and on those processes adaptation is worth a median $\mathcal{A}$ of 0.7% to 1.8% on the inventory problem, across both ends of the risk-aversion band and both mappings from data to the impact coefficient that the linear-impact model admits, the Amihud ratio (Amihud, 2002), which averages absolute return over dollar volume, and volatility over dollar volume. On a call option whose strike is 10% above the stock price, with ten days to expiration it is worth a median 13.0%, which is the largest cell on a surface over maturities and strikes. The ordering is the one we find in Section 4 between a regime that moves the target and one that moves only the costs, in $\mathcal{A}$ here and in $\mathcal{V}$ there. The inventory problem is the small one because a volatility regime is also an illiquidity regime for these stocks, and higher volatility speeds the optimal trading rate while a higher cost of trading slows it, so the two regimes' schedules nearly agree. Since s cannot exceed one, by Equation (5) any improvement above 1.8% on the median stock is baseline shortfall rather than adaptation. The two problems therefore differ by roughly an order of magnitude in what adaptation is worth, and no algorithm changes between them. In Appendices C and D, we give the $\mathcal{A}$ surfaces that these medians summarize, the mechanism and the sensitivity checks.

8 Discussion and Conclusion

A relative improvement mixes what the environment had to give, the fraction the agent took, and how far the baseline fell short, and only the second belongs to the algorithm. Separating the three makes such a number comparable across environments, since the other two move with the environment and the baseline. The bound needs only a number a paper already reports, and the two-run construction needs no model either, only one algorithm trained twice on one environment. Computing $\mathcal{A}$ and s needs our class, which is why the suite is more important than any number in it.

Computability costs three assumptions, that the regime is exogenous, that it moves the cost and the observation law but never the dynamics, and that it recurs with fixed statistics throughout training, so our agent results say nothing about a change met once. A regime in the dynamics would give the action a dual effect (Bar-Shalom & Tse, 1974). Outside the class our evidence is the latent-mode process of Section 4 and the POBAX domains. We draw only long-lived stocks, and so select survivors. Four of the seven algorithms fail the validity check on one problem or the other, so the cost-target evidence rests on four (Table 1), and on the calibrated configurations no algorithm reaches either reference (Appendix J). We build a benchmark that supplies the three reference values, so an algorithm can state the fraction it collected rather than the improvement it reported.

Reproducibility Statement

Every number and figure in this paper comes from a committed script that reads cached experiment outputs, and the three reference values follow from the model parameters alone, so reproducing them requires no training. The POBAX runs of Section 6 are the only ones that need a GPU, and in Appendix K we list what we commit, so both the accounting and our scoring choice can be checked without rerunning them. We fixed the hypotheses and thresholds of the three studies, of the bootstrapping prediction and of the calibrated-agent study before the corresponding runs, we record every verdict in Table 1, and in Appendix B.5 we set out the choices we made after seeing results. In Appendix B, we give the environment and controller construction, the checks that match each controller to the agents' environment, the configuration grids and all hyperparameters. The released suite is an installable Python package, supplied for review in the supplementary material and deposited publicly on acceptance, through which we expose its 21 configurations with `make_env` and `reference_values` and supply the 496 fitted regime processes.

AI Usage Statement

We used large language model assistants (Claude, Anthropic) throughout this project, in three broad roles. They contributed to the research design: the conceptual framework, the formal claims and their derivations, the hypotheses we fixed in advance, and the experiments and their validity checks. They built and ran the software: the environments, controllers and agents, the simulated regime paths, the preparation of the market data, the cluster runs, and the collection and analysis of results. They assisted with reporting: the figures, the tables, the text and the literature search, and with interpreting results. We did not use them to translate, and qualitative or thematic data analysis does not apply to this work. We reviewed all AI-assisted work, revised it where needed, and integrated only what we had checked. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of these assistants.

References

Rishabh Agarwal, Marlos C. Machado, Pablo Samuel Castro, and Marc G. Bellemare. Contrastive behavioral similarity embeddings for generalization in reinforcement learning. In *International Conference on Learning Representations*, 2021a.

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

Cameron Allen, Aaron Kirtland, Ruo Yu Tao, Sam Lobel, Daniel Scott, Nicholas Petrocelli, Omer Gottesman, Ronald Parr, Michael L. Littman, and George Konidaris. Mitigating partial observability in sequential decision processes via the lambda discrepancy. In *Advances in Neural Information Processing Systems 37*, 2024. doi: 10.52202/079017-2011.

Robert Almgren and Neil Chriss. Optimal execution of portfolio transactions. *The Journal of Risk*, 3 (2):5–39, 2001. doi: 10.21314/JOR.2001.041.

Yakov Amihud. Illiquidity and stock returns: Cross-section and time-series effects. *Journal of Financial Markets*, 5(1):31–56, 2002. doi: 10.1016/S1386-4181(01)00024-6.

Karl Johan Åström. Optimal control of Markov processes with incomplete state information. *Journal of Mathematical Analysis and Applications*, 10:174–205, 1965. doi: 10.1016/0022-247X(65)90154-X.

Yaakov Bar-Shalom and Edison Tse. Dual effect, certainty equivalence, and separation in stochastic control. *IEEE Transactions on Automatic Control*, 19(5):494–500, 1974. doi: 10.1109/TAC.1974.1100635.

Fischer Black and Myron Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(3):637–654, 1973. doi: 10.1086/260062.

W. P. Blair and D. D. Sworder. Feedback control of a class of linear discrete systems with jump parameters and quadratic cost criteria. *International Journal of Control*, 21(5):833–841, 1975. doi: 10.1080/00207177508922037.

Alessio Brini and Petter N. Kolm. On deep reinforcement learning for dynamic trading with PPO: Challenges and future directions. *The Journal of Financial Data Science*, 8(2):31–52, 2026. doi: 10.3905/jfds.2026.001.

Wang Chi Cheung, David Simchi-Levi, and Ruihao Zhu. Reinforcement learning for non-stationary Markov decision processes: The blessing of (more) optimism. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 1843–1854, 2020.

H. J. Chizeck, A. S. Willsky, and D. Castanon. Discrete-time Markovian-jump linear quadratic optimal control. *International Journal of Control*, 43(1):213–231, 1986. doi: 10.1080/00207178608933459.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 1724–1734, 2014. doi: 10.3115/v1/D14-1179.

Leilei Cui, Zhong-Ping Jiang, Petter N. Kolm, and Grégoire G. Macqueron. A fully data-driven value iteration for stochastic LQR: Convergence, robustness, and stability. *IEEE Transactions on Neural Networks and Learning Systems*, 2026. doi: 10.1109/TNNLS.2026.3675892.

Michel De Lara and Olivier Gossner. Payoffs-beliefs duality and the value of information. *SIAM Journal on Optimization*, 30(1):464–489, 2020. doi: 10.1137/18M1230049.

Sagar Deb and Ashwanth Krishnan. STOCKTAKE: Measuring the gap between perception and action in LLM agents with a fair oracle. *arXiv preprint arXiv:2607.13618*, 2026.

Yuhao Ding, Ming Jin, and Javad Lavaei. Non-stationary risk-sensitive reinforcement learning: Near-optimal dynamic regret, adaptive detection, and separation design. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37:7405–7413, 2023. doi: 10.1609/aaai.v37i6.25901.

Yan Duan, John Schulman, Xi Chen, Peter L. Bartlett, Ilya Sutskever, and Pieter Abbeel. RL^2: Fast reinforcement learning via slow reinforcement learning. *arXiv preprint arXiv:1611.02779*, 2016.

Michael O. Duff. *Optimal Learning: Computational Procedures for Bayes-Adaptive Markov Decision Processes*. PhD thesis, University of Massachusetts Amherst, 2002.

Daniel Ebi, Damien Ernst, Klemens Böhm, and Gaspard Lambrechts. Informed asymmetric actor-critic: Leveraging privileged signals beyond full-state access. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Fan Feng, Biwei Huang, Kun Zhang, and Sara Magliacane. Factored adaptation for non-stationary reinforcement learning. In *Advances in Neural Information Processing Systems 35*, 2022.

Songtao Feng, Ming Yin, Ruiquan Huang, Yu-Xiang Wang, Jing Yang, and Yingbin Liang. Non-stationary reinforcement learning under general function approximation. In *International Conference on Machine Learning*, 2023.

Scott Fujimoto, Herke van Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 1587–1596. PMLR, 2018.

Audrunas Gruslys, Will Dabney, Mohammad Gheshlaghi Azar, Bilal Piot, Marc Bellemare, and Remi Munos. The reactor: A fast and sample-efficient actor-critic agent for reinforcement learning. In *International Conference on Learning Representations*, 2018.

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*, pp. 1861–1870. PMLR, 2018.

James D. Hamilton. A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica*, 57(2):357–384, 1989. doi: 10.2307/1912559.

Milos Hauskrecht. Value-function approximations for partially observable markov decision processes. *Journal of Artificial Intelligence Research*, 13:33–94, 2000. doi: 10.1613/jair.678.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. doi: 10.1609/aaai.v32i1.11694.

Ronald A. Howard. Information value theory. *IEEE Transactions on Systems Science and Cybernetics*, 2(1):22–26, 1966. doi: 10.1109/TSSC.1966.300074.

Sinan Ibrahim, Grégoire Ouerdane, Hadi Salloum, Henni Ouerdane, Stefan Streif, and Pavel Osinenko. Benchmarking reinforcement learning via stochastic converse optimality: Generating systems with known optimal policies. *arXiv preprint arXiv:2603.17631*, 2026.

Scott M. Jordan, Yash Chandak, Daniel Cohen, Mengxue Zhang, and Philip S. Thomas. Evaluating the performance of reinforcement learning algorithms. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

Khimya Khetarpal, Matthew Riemer, Irina Rish, and Doina Precup. Towards continual reinforcement learning: A review and perspectives. *Journal of Artificial Intelligence Research*, 75:1401–1476, 2022. doi: 10.1613/jair.1.13673.

Erwan Lecarpentier, David Abel, Kavosh Asadi, Yuu Jinnai, Emmanuel Rachelson, and Michael L. Littman. Lipschitz lifelong reinforcement learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35:8270–8278, 2021. doi: 10.1609/aaai.v35i9.17006.

Bruce D. Lee, Leonardo F. Toso, Thomas T. Zhang, James Anderson, and Nikolai Matni. Regret analysis of multi-task representation learning for linear-quadratic adaptive control. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39:18062–18070, 2025. doi: 10.1609/aaai.v39i17.33987.

Long-Fei Li, Peng Zhao, and Zhi-Hua Zhou. Dynamic regret of adversarial MDPs with unknown transition and linear function approximation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38:13572–13580, 2024. doi: 10.1609/aaai.v38i12.29261.

Arun Mallya and Svetlana Lazebnik. PackNet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7765–7773, 2018. doi: 10.1109/CVPR.2018.00810.

Weichao Mao, Kaiqing Zhang, Ruihao Zhu, David Simchi-Levi, and Tamer Başar. Near-optimal model-free reinforcement learning in non-stationary episodic MDPs. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

Golnaz Mesbahi, Parham Mohammad Panahi, Olya Mastikhina, Steven Tang, Martha White, and Adam White. Position: Lifetime tuning is incompatible with continual reinforcement learning. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

Vasanttilak Naik. Option valuation and hedging strategies with jumps in the volatility of asset returns. *The Journal of Finance*, 48(5):1969–1984, 1993. doi: 10.1111/j.1540-6261.1993.tb05137.x.

Tianwei Ni, Benjamin Eysenbach, and Ruslan Salakhutdinov. Recurrent model-free RL can be a strong baseline for many POMDPs. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 16691–16723. PMLR, 2022.

Masaki Ogura, Ahmet Cetinkaya, Tomohisa Hayakawa, and Victor M. Preciado. State-feedback control of Markov jump linear systems with hidden-Markov mode observation. *Automatica*, 89:65–72, 2018. doi: 10.1016/j.automatica.2017.11.022.

Ian Osband, Yotam Doron, Matteo Hessel, John Aslanides, Eren Sezener, Andre Saraiva, Katrina McKinney, Tor Lattimore, Csaba Szepesvári, Satinder Singh, Benjamin Van Roy, Richard Sutton, David Silver, and Hado Van Hasselt. Behaviour suite for reinforcement learning. In *International Conference on Learning Representations*, 2020.

Sindhu Padakandla. A survey of reinforcement learning algorithms for dynamically varying environments. *ACM Computing Surveys*, 54(6):1–25, 2021. doi: 10.1145/3459991.

Emilio Parisotto, Francis Song, Jack Rae, Razvan Pascanu, Caglar Gulcehre, Siddhant Jayakumar, Max Jaderberg, Raphael Lopez Kaufman, Aidan Clark, Seb Noury, Matthew Botvinick, Nicolas Heess, and Raia Hadsell. Stabilizing transformers for reinforcement learning. In *International Conference on Machine Learning*, volume 119, pp. 7487–7498, 2020.

Andrew Patterson, Samuel Neumann, Martha White, and Adam White. Empirical design in reinforcement learning. *Journal of Machine Learning Research*, 2024.

Riccardo Poiani, Andrea Tirinzoni, and Marcello Restelli. Meta-reinforcement learning by tracking task non-stationarity. In *International Joint Conference on Artificial Intelligence*, 2021.

Raghu Rajan, Jessica Lizeth Borja Diaz, Suresh Guttikonda, Fabio Ferreira, André Biedenkapp, Jan Ole von Hartz, and Frank Hutter. MDP playground: An analysis and debug testbed for reinforcement learning. *Journal of Artificial Intelligence Research*, 77:821–890, 2023. doi: 10.1613/jair.1.14314.

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*, pp. 5331–5340. PMLR, 2019.

A Run and Ziluo Ding. In-context reinforcement learning under non-stationarity: A survey. *arXiv preprint arXiv:2607.11906*, 2026.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *International Conference on Learning Representations*, 2016.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Xingyou Song, Yiding Jiang, Stephen Tu, Yilun Du, and Behnam Neyshabur. Observational overfitting in reinforcement learning. In *International Conference on Learning Representations*, 2020.

Ruo Yu Tao, Kaicheng Guo, Cameron Allen, and George Konidaris. Benchmarking partial observability in reinforcement learning with a suite of memory-improvable domains. *Reinforcement Learning Journal*, 6:1412–1439, 2025.

Guy Tennenholtz, Uri Shalit, and Shie Mannor. Off-policy evaluation in partially observable environments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 10276–10283, 2020. doi: 10.1609/aaai.v34i06.6590.

Ardianto Wibowo, Paulo E. Santos, Amer Baghdadi, Matthew Stephenson, Karl Sammut, and Jean-Philippe Diguet. A unified causal-origin taxonomy of distributional shifts in reinforcement learning. *arXiv preprint arXiv:2606.16933*, 2026.

Maciej Wołczyk, Michał Zając, Razvan Pascanu, Łukasz Kuciński, and Piotr Miłoś. Continual world: A robotic benchmark for continual reinforcement learning. In *Advances in Neural Information Processing Systems 34*, pp. 28496–28510, 2021.

Amy Zhang, Shagun Sodhani, Khimya Khetarpal, and Joelle Pineau. Learning robust state abstractions for hidden-parameter block MDPs. In *International Conference on Learning Representations*, 2021.

Xiang Zhou, Yi Xiong, Ningyuan Chen, and Xuefeng Gao. Regime switching bandits. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

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.

A THE BACKWARD RECURSION

Both controllers run the same recursion on the stage cost of Equation (1) under different information, so we state it once. The recursion over regimes is the backward induction for a discrete-time Markov jump linear system with a quadratic cost and perfect observation of the mode. Chizeck et al. (1986) give the coupled Riccati-like matrix difference equations from which the optimal control laws and expected costs-to-go can be precomputed, and Blair & Sworder (1975) give an early treatment of this class. The stage cost here carries a drift and a regime-dependent target, so our recursion also carries the affine coefficients β_t and γ_t, which is a routine extension. The same recursion over a grid of beliefs is a different object, an approximation to a partially observable problem on a discretized simplex.

We write the value of holding state x at step t under index i as $J_t(x, i) = \alpha_t[i]\, x^2 + \beta_t[i]\, x + \gamma_t[i]$, where i ranges over regimes for the regime-informed controller and over a grid of beliefs for the belief-optimal one. The episode ends by returning the state to zero and paying only the cost of acting, which requires a last action of $-d/a$ and gives $\alpha_{T-1} = c$, $\beta_{T-1} = 2cd/a$ and $\gamma_{T-1} = cd^2/a^2$. Every configuration we run sets the drift d to zero, where the last two vanish. We write K for the transition kernel over the index, which is P for the regime-informed controller and the belief transition induced by the observation for the belief-optimal one, and abbreviate $\bar\alpha = K\alpha_{t+1}$, $\bar\beta = K\beta_{t+1}$, $\bar\gamma = K\gamma_{t+1}$. Then with

$$\Gamma_t = c + \rho + a^2\bar\alpha, \qquad g_t = \rho\, m - a\, d\, \bar\alpha - \tfrac{1}{2} a\bar\beta, \tag{7}$$

the coefficients at step t are

$$\alpha_t = \frac{c\,(\rho + a^2\bar\alpha)}{\Gamma_t}, \qquad \beta_t = \frac{-2c\, g_t}{\Gamma_t}, \qquad \gamma_t = \rho\, m^2 + d^2\bar\alpha + d\bar\beta + \bar\gamma - \frac{g_t^2}{\Gamma_t}, \tag{8}$$

and the optimal action is affine in the state, $u_t^\star(x) = (c/\Gamma_t)\, x + g_t/\Gamma_t$, so it is a gain and a bias per index and per step.

The two controllers differ only in what indexes the recursion. The regime-informed controller runs it over the N regimes with $K = P$ and the per-regime parameters $(c_z, \rho_z, \rho_z m_z, \rho_z m_z^2)$. The belief-optimal controller runs it over a grid of beliefs, with those four quantities replaced by their belief averages and K replaced by the belief transition. The observation law depends on the regime alone, so the belief update does not depend on the action, there is no dual effect (Bar-Shalom & Tse, 1974), and the reduction to a fully observed problem on (x_t, b_t) is exact rather than an approximation. An agent here cannot act in order to learn about the regime. The state trajectory carries no evidence about the regime either, since the regime is absent from the dynamics, so inference runs on the observation stream alone.

We keep ρm and ρm^2 as separate belief averages rather than collapsing them into a single averaged target. Only γ_t reads ρm^2, so the collapse leaves α_t, β_t, the gain and the bias unchanged at every step, and the optimal policy with them. It changes the reported cost alone, which falls by the belief average of ρm^2 minus $(\overline{\rho m})^2/\bar\rho$, accumulated backward through the kernel. That quantity is the belief variance of the target weighted by the tracking parameter, and it is nonnegative by the Cauchy-Schwarz inequality. It vanishes for the regime-informed controller, whose index is a single regime. The collapse would therefore lower J^{bel}, leave J^{inf} where it is, and bias $\mathcal{V}$ downward.

The no-information bound is the same recursion with the belief frozen at the chain's stationary distribution, so its kernel is the identity on one point.

Justification of Remark 3.1. To see it, write $1 - \mathcal{V} = J^{\mathrm{inf}}/J^{\mathrm{bel}}$ and $1 - \mathcal{A} = J^{\mathrm{bel}}/J^{\mathrm{noinf}}$, so J^{bel} cancels in the product and gives Equation (6). The regime-informed and uninformed controllers both act without filtering, so J^{inf} and J^{noinf} do not move and their ratio is fixed. With both of them fixed and positive, $1 - \mathcal{V} = J^{\mathrm{inf}}/J^{\mathrm{bel}}$ falls and $1 - \mathcal{A} = J^{\mathrm{bel}}/J^{\mathrm{noinf}}$ rises in J^{bel}, so $\mathcal{V}$ increases in it while $\mathcal{A}$ decreases.

B Implementation Details

B.1 Environment

The agent-facing environment wraps the scalar control problem of Section 3 in a high-dimensional input so that a deep policy has to learn a representation. The controllable state is two dimensional, the fraction of the horizon remaining and the scalar state x_t of Equation (1), which is the fraction of inventory remaining in the inventory problem and the tracked position in the tracking problems, and the agent's input is a fixed random two-layer embedding of that state into $\mathbb{R}^{64}$, concatenated with a window of the most recent observations so the regime can be inferred from the observation stream rather than handed over. We follow Song et al. (2020) in projecting a low-dimensional control state through a fixed random map into a higher-dimensional observation. Projected Gym applies a linear semi-orthogonal projection. Ours is a fixed two-layer tanh network, so we take the family from that work and the map itself is our design choice. It carries the two-dimensional state through a hidden layer of width 32 into 64 dimensions with a tanh at each layer, and we draw its weights once from $\mathcal{N}(0, 1)$ into the hidden layer and $\mathcal{N}(0, 1/32)$ out of it, its biases from $\mathcal{N}(0, 0.3^2)$, under a seed that forms part of each configuration's identity. We run no agent on the raw two-dimensional state, so the embedding fixes what the policy has to decode rather than measuring how hard the problem is without it. We run every agent study over 26 steps per episode with a 10-observation window unless we state otherwise. The cost parameters, the step length, the risk aversion, and the regime process are per study and appear in Appendix B.2. In the informed condition we append the true regime to the input as a one-hot and change nothing else.

B.2 Configuration grids

We give here the numeric grid behind every study in the paper, so each reported configuration is reproducible from its parameters alone. For the three components of Section 4, all 134 configurations run over a 20-step horizon from the common starting state $x_0 = 0$, with observation standard deviation 0.02 in both regimes, observation separation 0.05 except in the observation component, and self-transition probabilities $(0.97, 0.92)$ except where a component varies them. The cost-weight component crosses the cost ratio c_1/c_0 with the tracking-weight ratio ρ_1/ρ_0, seven geometric levels each from 1 to 16, with the target at 0.5 in both regimes, for 49 configurations. The cost-target component crosses the target separation, seven levels from 0 to 0.8, with the common cost level, seven geometric levels from 0.5 to 8, for 49. The observation component fixes $c = (2, 2)$, $\rho = (1, 1)$, and targets $(0.2, 0.8)$, then crosses the observation separation, twelve levels from 0.002 to 0.12, with three self-transition settings, 0.99, 0.97, and 0.90 applied to both regimes, for 36.

The crossed grid, where we settle the component comparison of Section 4, is separate from those 134 and holds fewer control settings, because crossing multiplies each level by a factor of seven. We run 98 configurations in it, the full product of the two components that carry a control parameter with seven observation separations. Only two components enter it, since the third is the observation component itself and cannot also be crossed against it. The cost-weight component varies the cost ratio over seven geometric levels from 1 to 16 at $\rho = (1, 1)$ and a common target of 0.5, and the cost-target component varies the target separation over seven levels from 0 to 0.8 at $c = (2, 2)$ and $\rho = (1, 1)$. We cross both components with the observation separation over seven geometric levels from 0.002 to 0.12. The observation standard deviation is 0.02 in both regimes, the self-transition probabilities are $(0.97, 0.92)$, the horizon is 20 steps from $x_0 = 0$, and the belief grid has 301 points with 601 quadrature nodes. Two of the fourteen control settings are degenerate, the cost-target component at zero target separation and the cost-weight component at unit cost ratio, whose regimes are identical and whose quantities are zero to machine precision.

The two coordinates are the displacement diagnostic and the logarithm of the observation separation. We compute the diagnostic as the standard deviation of the regime-informed controller's first action across regimes at the shared starting state, weighted by how often each regime occurs. The reported 64.8% is the coefficient of determination of an ordinary least-squares regression of $\mathcal{V}$ on those two with an intercept, over all 98 configurations. We add a binary indicator for the cost-target component and it rises to 72.1%, an increment of 7.3 percentage points. We apply no transform to $\mathcal{V}$ and the two coordinates enter linearly.

We build the first coordinate on De Lara & Gossner (2020), whose sign condition on the optimal actions at the prior belief leaves open whether it orders a sequential problem. It ranks the two cost components almost as $\mathcal{V}$ does, at 0.985 and 0.991, and reaches only 0.445 on the observation component, because it reads a controller that never reads the observation.

For the inventory grid of Section 7, the 576,000 configurations are the product of six axes: the volatility ratio and the impact ratio, twenty geometric levels each from 1 to 25, the calm self-transition probability over $\{0.90, 0.95, 0.97, 0.98, 0.99, 0.995\}$, the turbulent one over $\{0.80, 0.88, 0.92, 0.95, 0.97, 0.99\}$, risk aversion over eight geometric levels from 10^2 to 3×10^5, and the horizon over $\{6, 13, 26, 39, 78\}$ steps. Calm volatility is 0.008, calm impact is 1, and the step length is 0.05 throughout. Two regimes' schedules agree within 5% when the ratio of their optimal first-step trading fractions lies in $[0.95, 1.05]$, which holds for 31,830 of the configurations.

In the inventory ranking study, we use twelve inventory configurations, each a (volatility ratio, impact ratio) pair on a $\sqrt{2}$ lattice: $(1, 1)$, $(5.66, 1.41)$, $(5.66, 1)$, $(8, 1)$, $(2.83, 11.31)$, $(1, 5.66)$, $(1.41, 8)$, $(1, 8)$, $(1.41, 11.31)$, $(1, 11.31)$, $(1.41, 16)$, and $(1, 16)$. Calm volatility is 0.008, calm impact is 1, the step length is 0.05, risk aversion is 10^4, and the self-transition probabilities are $(0.97, 0.92)$, with eight seeds in each configuration and condition.

We run the identifiability study and the memory study on one control problem throughout, with cost of acting $c = (1, 4)$, tracking weight $\rho = (1, 1)$, unit step length and unit risk aversion, started at $x_0 = 1$, with self-transition probabilities $(0.97, 0.92)$ and observation standard deviation 0.02 in both regimes. In the identifiability study, we cross six observation separations, 0.002, 0.006, 0.012, 0.020, 0.035, and 0.060, at eight seeds per condition. The memory study, the generalized advantage estimation (GAE; Schulman et al., 2016) λ axis of Appendix E, and the two off-policy variants use the four separations 0.002, 0.012, 0.020, and 0.060 at six seeds per condition. The pooled on-policy count of 72 is the three PPO variants over those four separations at six seeds, with the masked condition held out, and the off-policy count of 48 is TD3 and SAC over the same grid.

In the target-separation study of Section 5, we use a different control problem, since the cost-target component is one the inventory shape cannot express. Costs are common to both regimes, $c = (2, 2)$ and $\rho = (1, 1)$, with unit step length and unit risk aversion, started at $x_0 = 0$ away from both targets. The targets are $0.5 \pm \Delta/2$ for five separations $\Delta \in \{0, 0.2, 0.4, 0.6, 0.8\}$, and the regime process is held fixed throughout at self-transition probabilities $(0.97, 0.92)$ with observations drawn from $\mathcal{N}(\pm 0.05, 0.02^2)$, so only the targets move across configurations. Feedforward PPO, recurrent PPO, SAC and TD3 each run at six seeds in both conditions, which is 240 runs, and the masked condition adds 30, the context encoder 60, and the four meta-reinforcement learning variants 240, for 570 runs in all, before the escalation of Appendix B.4 adds 960 more across its four algorithms. The 28 agent-environment pairs we decompose in Section 6 are the four algorithms that pass the validity check at each of the four non-zero target separations, together with the three PPO variants of the memory study at each of its four observation separations. The action is the new position itself, read as a displacement from the common reference position 0.5, rather than the fraction of remaining inventory the inventory studies use.

B.3 Controllers and their matching to the agents' environment

Two tests stand behind the claim that these controllers are optimal. We roll out the regime-informed controller and require that the simulated cost match the cost its own backward recursion predicts, and we perturb the actions that recursion computes and require that no perturbation lower the simulated cost.

We compute the regime-informed controller by backward induction over regimes, and the belief-optimal controller by backward induction over a discretized belief space where we integrate over next observations with a fine midpoint rule. Gauss quadrature is the wrong tool here even though the observation is Gaussian, because the integrand is the posterior as a function of the observation, which saturates toward zero and one and behaves almost like a step. We use belief grids of 121 to 401 points with 301 to 801 quadrature nodes across studies, and we place the midpoints over ten standard deviations either side of each regime's own observation mean, on that regime's own scale, so that we resolve the sharp regime and the diffuse one alike. The coarsest carries the 576,000-configuration inventory grid and the finest carries the agent studies. We refine the belief grid through 101, 201 and 401 points with 401, 801 and 1,601 quadrature nodes and require the discretization error against an

exactly known reference to fall by more than a factor of three at each doubling, which it does. The direction of that error is as important as its size, because we read the belief-optimal cost as a bound in Section 3. The recursion returns each successor belief to the grid by splitting its mass between the two neighboring points in proportion to distance, which preserves the belief's mean and replaces a point by a spread around it, so the sign follows from the curvature of the cost-to-go. Refining through 1,601 points settles the direction. The cost increases at every refinement on every configuration where the regimes differ, so the computed value is a lower bound on the true one. That direction errs against the agent, since no feasible policy improves on the true value, and so no policy improves on a value below it. At the 401 points the agent studies use, that gap reaches 0.007 basis points, a millionth of cost, which is four orders of magnitude below the Monte Carlo band reported next and two below the smallest anomaly we record in this appendix.

Two matching conditions align the controller's starting belief and cost timing with the agent's. The agent begins each episode with a window of past observations, so the belief-optimal controller's initial belief is drawn from the distribution that a burn-in of that length induces rather than from the stationary prior, and we estimate that distribution by Monte Carlo over 200,000 paths. That estimate carries a sampling error not covered by the grid refinement above, and it is the larger of the two. Across twenty burn-in seeds on the most identifiable observation configuration, where the reference value is smallest and the relative error therefore largest, $\mathcal{V}$ runs from 1.44% to 1.74%, a spread of 0.30 percentage points and a standard deviation of 0.08. Any comparison finer than that band is inside the reference value's own noise, which is why we do not read agent differences below it. We fix the released suite's burn-in seed as part of each configuration's identity, so its values are reproducible, and we reproduce the inventory ranking, target-separation and memory studies from it exactly. We run the identifiability study on the memory study's control problem at four of its six observation separations and draw its burn-in with its own seed, so its values differ from the released entries by 0.04 to 0.19 percentage points, inside the band. And the environment charges the cost of step t under the regime at t while it shows observations only through $t-1$, so the controller's stage-cost weights are the one-step-ahead prediction of its belief.

We apply both in every study. The error from either convention is worth several percent of cost, which is larger than most of the effects we measure in Section 5. The two pull in opposite directions, since a burn-in belief is sharper than the prior while a predicted one is blurrier. On the inventory ranking study's twelve configurations, applying only the first moves $\mathcal{V}$ by up to 5.4 percentage points and applying both moves it by at most 1.6. A sign check decides between applying only the burn-in belief and applying both, since without either the agent appears to beat its own belief-optimal controller by 0.02% on one configuration, which is impossible for a feasible policy.

One gap remains by construction, namely that the belief-optimal controller conditions on the whole observation history while a feedforward agent sees a fixed window, so the controller's information set is slightly stronger than that agent's. In our parameterization the difference is negligible, at a mean belief error of 0.3848 against 0.3865 in the hardest inference setting and identical to four decimals elsewhere.

B.4 Agents

A feedforward policy sees only a fixed window, so it cannot run the controller's filter and its shortfall is confounded with that limit, while a recurrent policy carries the whole history, so a shortfall that survives there is not an artifact of the window, and we therefore run both.

Every agent with a stochastic policy shares the policy class, an unbounded Gaussian that the environment rather than the policy maps onto the problem's own action. Under tracking that map is an affine shift by a reference position common to both regimes, and it is unbounded. Under liquidation it is a sigmoid onto the fraction of remaining inventory, and the optimal fraction lies inside its image at every step the agent controls. Both maps keep the problem in the linear-quadratic class, so the controllers of Appendix A remain the optima of the problem the agent faces. TD3 carries a deterministic actor instead, which is why the unbounded action costs it more than the others. They also share the evaluation protocol, 8,192 evaluation episodes per run, with the seed count per study given in Appendix B.2.

Every interval we report is a percentile interval over seeds. A run's seed is the unit of replication, since it is the only thing that varies within a configuration and a condition, and we draw seeds

with replacement, recompute the statistic on each draw, and take the 2.5th and 97.5th percentiles of 20,000 draws, except in Table A2 and for the meta-reinforcement-learning variants, where we take 10,000. We use percentile intervals rather than a normal approximation because the statistics are ratios and rank correlations, whose sampling distributions we have no reason to treat as symmetric, and because at 6 or 8 seeds an interval built from an estimated standard error would rest on that symmetry. A measured gap is a ratio of the two conditions' mean costs, so we resample each condition on its own, while an inference penalty is a difference of two shortfalls, so we match the conditions by configuration and seed and resample the paired differences. Where a statistic runs across configurations, as the inventory ranking study's rank correlation does, one draw per configuration enters each replicate, so the interval carries the measurement noise in every point at once. Each analysis script fixes its own resampling seed, so the intervals reproduce exactly. The escalation to 30 seeds is a two-stage rule, so an interval reported after it is conditional on the first stage having covered the threshold and its coverage is not the nominal one. We escalate to avoid reading a verdict off an interval that does not resolve, rather than to calibrate a test.

We fix hyperparameters across conditions and configurations within a study (Table A1). We did not tune PPO, since its learning rate of 3×10^{-4}, clip ratio of 0.2 and GAE λ of 0.95 are the continuous-control settings Schulman et al. (2017) report, and we run four optimization epochs over four minibatches where that paper runs ten epochs. We fix all of them once and hold them across every configuration and both conditions, and we never revisit one after seeing a result. The only PPO setting a study chooses is the optimization budget, and we choose it at the configuration where the regime carries no information, so the choice cannot favor either condition. Tuning per configuration would make the measured gap a property of our search rather than of the environment, which is the quantity we read in Section 5. SAC and TD3 carry one budget selection each, described below, chosen once and globally rather than per configuration.

Table A1: Hyperparameters for the three base algorithms, held fixed across every configuration and across the uninformed and informed conditions within a study. A dash marks a hyperparameter that does not apply to that algorithm. Three rows vary by study rather than by configuration: the two optimization-budget rows, since the cost-target problem takes the larger budget, selected at zero target separation where the regime carries no information, and the GAE λ row, which the λ axis of Appendix E varies on purpose. The meta-reinforcement-learning variants and the context encoder are PPO policies at their own selected budgets, reported in Appendix H, and take every other setting from the PPO column.

	PPO	SAC	TD3
hidden units per layer	64	64	64
iterations, inventory	400	300	200
iterations, cost-target	1200	900	600
episodes per iteration	512	64	64
learning rate	3×10^{-4}	3×10^{-4}	3×10^{-4}
gradient clip norm	0.5	0.5	0.5
clip ratio	0.2	—	—
optimization epochs $\times$ minibatches	4×4	—	—
GAE λ	0.95	—	—
entropy coefficient	3×10^{-3}	auto, target -1	—
gradient steps per iteration	—	200	200
batch size	—	256	256
replay buffer	—	400,000	400,000
target update rate	—	0.005	0.005
exploration noise	—	learned	0.3
target policy noise (clip)	—	—	0.2 (0.5)
policy delay	—	—	2

We set the four thresholds in Table 1 at round levels rather than deriving them, so what defends them is where the verdicts fall and how far their intervals reach. Under the informed-run validity check at 15%, we admit an algorithm whose informed evaluation cost comes within 15% of the regime-informed controller's. An algorithm's verdict is the largest shortfall it reaches across the five target separations, since that is the configuration we read for the check, and in Table A2 we give every algorithm's deciding shortfall with a percentile bootstrap over seeds beside the verdict it receives at

three tolerances. Feedforward PPO, SAC and RL^2 given the action and reward pass at all three, and TD3 and RL^2 given the action only fail at all three. Four verdicts carried intervals that covered the 15% line at six seeds, and we ran each to 30 under the rule below: recurrent PPO, both VariBAD variants and the context encoder. The escalation left every verdict where it was and sharpened three of them. VariBAD given the action only falls from 16.3% to 10.8% and passes at all three tolerances, because its six-seed reading rested on one separation. VariBAD given the action and reward rises from 15.2% to 22.7% and now fails at 20% as well, and the context encoder's interval no longer reaches toward the line. The zero-value control at 1% is tighter because it reads a quantity that is zero by construction rather than a distance to an optimum. Three of the four passing algorithms read between 0.02% and 0.66%, and recurrent PPO reads -4.0% on an interval from -10.3% to 1.1%, which two of its thirty seeds carry against a median gap of 0.2%. The rank threshold at 0.7 carries the ranking hypotheses of the inventory ranking and target-separation studies, whose outcomes are 0.993 with bootstrap interval $[0.986, 1.000]$ and 1.000, and the identifiability study's hypothesis, whose outcome is -0.257. We fixed one rule in advance for the case where a level could decide an outcome, that a verdict whose bootstrap interval straddles its threshold raises the seed count to 30, and the four verdicts above invoked it.

Table A2: Informed-run validity verdicts against the tolerance that produces them, on the cost-target component. The deciding shortfall, as a percentage of the regime-informed controller's cost, is the largest an algorithm reaches across the five target separations, which is the quantity the check reads. Intervals are percentile bootstraps over seeds, 10,000 resamples, taken on that maximum. A dagger marks an algorithm averaged over 30 seeds rather than six, which we require wherever the six-seed interval covers the tolerance. TD3's cost diverges, so we omit its interval.

Algorithm	Deciding shortfall	95% interval	12%	15%	20%
SAC	2.0%	[1.7, 4.2]	pass	pass	pass
PPO, feedforward	7.6%	[5.1, 10.5]	pass	pass	pass
RL^2, action and reward	7.9%	[6.6, 13.9]	pass	pass	pass
PPO, recurrent†	9.2%	[8.5, 15.5]	pass	pass	pass
VariBAD, action only†	10.8%	[9.7, 15.1]	pass	pass	pass
VariBAD, action and reward†	22.7%	[17.6, 32.7]	fail	fail	fail
Context encoder†	59.6%	[39.6, 84.3]	fail	fail	fail
RL^2, action only	98.5%	[17.4, 254.3]	fail	fail	fail
TD3	diverges		fail	fail	fail

The feedforward policy is a two-layer tanh network over the input. The recurrent policy encodes the input and runs a gated recurrent unit (GRU) (Cho et al., 2014) across the full episode. We train it by backpropagation through the whole sequence, with minibatches formed over episodes rather than timesteps so the hidden state is never broken. We selected the budget for SAC and TD3 each once, globally, on three seeds and on the informed condition only, and selected the configuration with the lowest worst-case shortfall against the regime-informed controller from a shared grid over iterations, gradient steps, and exploration noise. All runs execute on CPU. A single run takes from under a minute on the inventory studies to 52 minutes on the longest calibrated configuration of the calibrated-agent study, and across the studies we report here we spend at least 1,650 core-hours over 3,600 runs, including the budget selections and the escalation. The validity check reads the informed condition, so that is where we record failures, though the uninformed condition is sometimes the worse of the two.

B.5 Scoring, tests, and choices made after seeing results

An agent's informed run scores against the regime-informed controller, its uninformed run against the belief-optimal controller's cost, and its masked run against the no-information bound. The informed run is the check a result must pass first. Our tests require that the released environment reproduce the one we trained in step by step, and that the suite reproduce the values the inventory ranking, target-separation and memory studies report. We run the identifiability study outside the suite, and on the configurations it holds in common with the memory study its regime-informed cost must match exactly while $\mathcal{V}$ must fall inside the band its burn-in belief draw induces.

We made the following choices after seeing results. We selected each algorithm's optimization budget at zero target separation, where the regime carries no information. We read two quantities there, how

close each condition comes to its own controller and how close the measured gap between them comes to zero, and the second decides where the two disagree. That gap is the quantity we test in the zero-value control row of Table 1 (Appendix H). The risk-aversion band uses a criterion fixed in advance, a first step in the low-volatility regime of 4 to 8 times the rate of the constant-rate schedule, which sells an equal share of the initial position at every step. We applied it after observing that low risk aversion implies slower than constant trading in the high-volatility regime, and we report the whole range. We conceived the comparison between inventory liquidation and hedging after the inventory result came back small. We chose the 10-day call 10% out of the money after computing the whole surface over maturities and moneyness, and it is one illustrative contract rather than a finding about that maturity or moneyness.

C THE INVENTORY CALIBRATION

Both calibrated problems reach Equation (1) through the same three coefficients, and we give that map here so the calibration reproduces from the fitted parameters alone. We write τ for the step length, ϕ for risk aversion, η_z for the temporary-impact coefficient and κ_z for the quadratic transaction cost. In the inventory problem the state is the position left to sell and the action is the position after trading, so the cost of acting is the impact coefficient over the step length, $c_z = \eta_z/\tau$, the tracking weight prices the risk carried into the next step, $\rho_z = \phi\,\sigma_z^2\tau$, and the target is $m_z = 0$ in both regimes. We set $\tau = 0.05$ throughout the inventory work, with the horizon as Appendix B.2 gives it. In the hedging problem the state is the stock holding, $c_z = \kappa_z$ and $\rho_z = \phi\,\sigma_z^2\tau$ again, with ϕ and τ both set to one so that the transaction cost carries that problem's one free scale, which Appendix D fixes at 10. The hedging problem takes the same fitted ratio for κ_1/κ_0 that the inventory problem takes for η_1/η_0, which is why one impact-ratio axis carries both. Its target m_z is the option's delta given regime z at inception, averaged over the chain's occupation time rather than taken at regime z's volatility, which would hold that regime to expiry and separate the two targets as far as they can go. The gain a is one and the drift d is zero in both. The volatility that prices the risk is the σ_z of Section 3, since here the observation is the return itself, and the impact ratio η_1/η_0 is the fitted Amihud ratio under the headline mapping and the fitted ratio of volatility to dollar volume under the second.

The 576,000-configuration surface is the product of the six axes in Appendix B.2. Two regimes' schedules agree within 5% when the ratio of their optimal first-step trading fractions lies in $[0.95, 1.05]$, which holds for 31,830 of those configurations, and across them $\mathcal{V}$ never exceeds 0.16%.

The 496 stocks come from a screen we fixed before fitting any model. We restrict to CRSP stocks with a long daily history between 2000 and 2024, rank them by average daily dollar volume, cut the ranking into ten equal groups, and draw 50 from each, so every drawn stock carries at least fourteen years of daily observations. We then keep a fit only when it converges with a positive volatility in each state and the smoothed probability assigns at least 100 days to each, and four of the 500 drawn stocks fail one of those conditions. Both requirements exist so that each state carries enough days for its volatility and its persistence to be identified. By requiring that history we select survivors, which we record in Section 8, and the sample size is not load-bearing, since every figure we report from these stocks is a cross-sectional summary rather than a test statistic.

We stratify the draw on dollar volume, which fixes the count in each decile at 50. The deciles partition the candidate universe into groups of equal size, so this has the same expected composition as a simple random draw and removes only the variation in how many stocks each decile contributes. Dollar volume is the axis we stratify on because it enters the impact coefficient directly under the linear-impact model, which makes dollar volume the input we most want represented at both ends of that coefficient's range. A median over these 496 stocks is therefore a median over a draw balanced across liquidity deciles rather than over the CRSP cross-section.

No liquidity variable enters the regime fit. We estimate it on returns alone, with a regime-dependent mean and variance, by maximum likelihood on the daily returns in percent, five expectation-maximization iterations from a default start followed by a quasi-Newton search. We then label the higher-volatility state turbulent, classify each day by whether the smoother puts its probability of that state above one half, and compare the liquidity measures between the two groups. If we defined the regimes on liquidity instead, the liquidity comparison would be circular, since the ratios we report would then be ratios across a split built from the same measures. We order the two steps this way so we can read those ratios as a statement about what liquidity does inside a volatility regime. We chose it before fitting anything, and a split defined on liquidity would place stocks differently, which is a modeling choice to test rather than assume.

Real stocks fall near the cancellation diagonal, the line in the volatility-ratio by impact-ratio plane where the two regimes' trading schedules agree, for a mechanical reason. Under the linear-impact model the impact coefficient is proportional to volatility over dollar volume. Dollar volume barely moves between a stock's regimes, at a median ratio of 0.95, so the volatility ratio divided by the dollar-volume ratio explains 97% of the cross-sectional variation in the Amihud ratio, whether the ratios enter at the level or in logs. A volatility regime is therefore an impact regime for these stocks, which we show rather than assume, and in Figure A1 we place the fitted stocks on the surface directly.

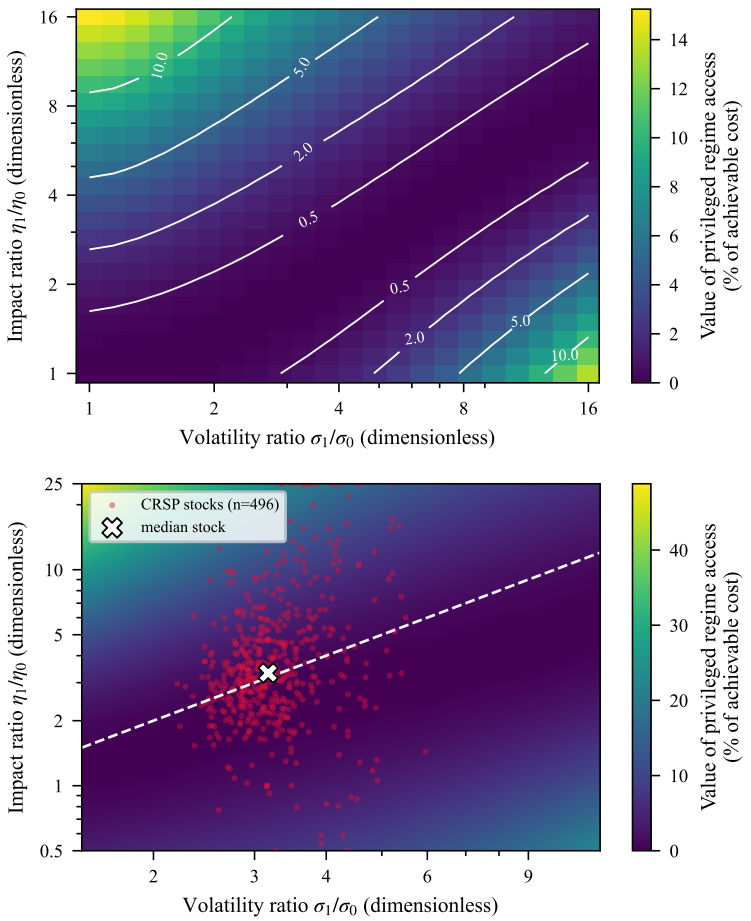


Figure A1: Top, value of privileged regime access for the inventory problem over the volatility-ratio by impact-ratio plane, where the impact ratio is η_1/η_0 and η_z is the temporary-impact coefficient, at 20 steps, risk aversion 5,000, persistence $(0.98, 0.95)$, with white contours at 0.5%, 2%, 5%, and 10%. Bottom, the same quantity at the horizon, risk aversion and median persistence of the CRSP fits, 78 steps, risk aversion 10,000 and $(0.966, 0.879)$, with the cancellation diagonal dashed in white, no contours, and the 496 stocks placed at their fitted volatility and Amihud ratios, the 19 whose Amihud ratio falls outside the plotted range drawn at its edge, and the median stock marked. Axes are logarithmic, and each panel carries its own color scale and ranges, with the lower one over the region the CRSP fits occupy.

Three mappings from returns and volume to an impact coefficient appear in practice. Two, the Amihud ratio and the volatility-over-volume ratio, are consistent with the linear-impact model and carry the headline band. The third, inverse dollar volume with no return in it, is not consistent with that model and puts the median of $\mathcal{V}$ at 2% to 5% and the median of $\mathcal{A}$ at 2.1% to 2.9%. We report both bands, and under the headline band the 90th percentile of $\mathcal{V}$ across the 496 stocks is 8% to 11%.

D HEDGING

We give here the hedging analysis behind the comparison of Section 7, on the same 496 stocks. They spread across the cancellation diagonal (Figure A2, drawn at 21 days) instead of occupying a corner of the parameter space. Their coordinates are the fitted volatility and Amihud ratios and so do not depend on maturity, while the levels do: on the 10-day contract 10% out of the money, 99.4% of them clear $\mathcal{V} = 5\%$.

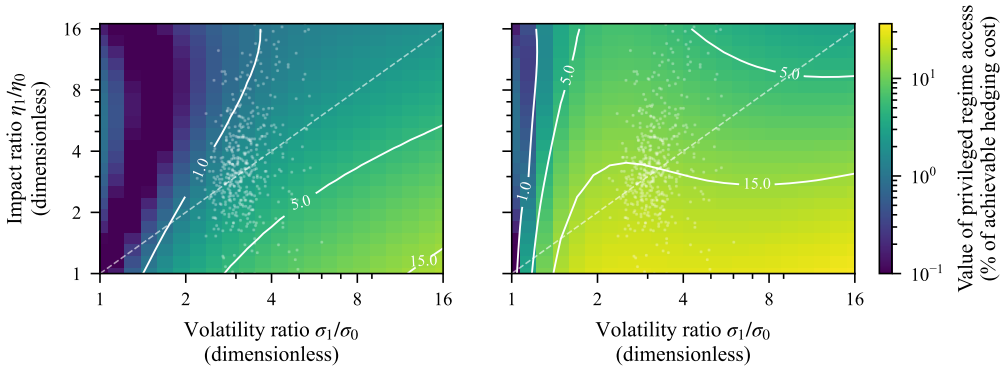


Figure A2: Value of privileged regime access for hedging a 21-day European call, over the volatility-ratio by impact-ratio plane at two strikes, with white contours at 1%, 5%, and 15% and the cancellation diagonal dashed. Left, at the money. Right, 10% out of the money. Points are the 496 CRSP stocks at their fitted volatility and Amihud ratios, 37 of which fall outside the plotted impact-ratio range and are not shown. Axes and color are logarithmic, with the color scale floored at 0.1%.

The hedging problem carries one free scale, the transaction cost relative to the per-step risk weight in the calm regime, and we set it to 10. It fixes how tightly the hedge tracks, and at that value the optimum closes about 27% of the tracking error per day, a partial rebalance rather than either a continuous hedge or a static one. Because it is a choice with no data behind it, we recompute the maturity by moneyness surface at 3 and at 30 as well, at the median calibration of the fitted stocks. On the 10-day contract 10% out of the money the level moves by roughly a factor of four across that range, and $\mathcal{V}$ still increases with moneyness at every maturity and every one of the three values, which is the comparison we read in Section 7. We compute the figures on the 496 stocks at 10 throughout.

We hold the hedging target fixed within an episode at the option's delta, and that delta moves for two reasons which bound the reported figures in opposite directions. Movement with the underlying price is common to all three controllers, so it inflates the denominator and makes every hedging fraction an upper bound on that channel. That movement cannot be priced in this class, because a target that moves with the spot makes the tracking error depend on a second state variable and leaves the class on which we can solve the controllers.

Movement with time to expiry is deterministic and therefore priceable. The delta of an out-of-the-money call decays toward zero as expiry approaches and decays faster under the calm regime than the turbulent one, so the two regimes' targets separate over the episode. We let the target follow that path, which keeps the recursion scalar, because the decay is known before the episode begins. On the 496 stocks the moving target raises the median $\mathcal{V}$ on the 10-day contract 10% out of the money from 15.2% to 22.0% and $\mathcal{A}$ from 13.0% to 14.2%, and it raises both by as much as 22 percentage points at 63 days. At the money, where the delta barely decays, neither moves. The body reports the fixed-target figures because they are the conservative choice on this channel. Since the body's target is held fixed within an episode, we draw no conclusion from how these figures vary with maturity.

E THE BOOTSTRAPPING AXIS

We difference an algorithm's two conditions to separate its cost of inferring the regime from whatever else it loses, and at the same observation window that difference puts TD3 above feedforward PPO by 0.62% of cost. We report that comparison in full below. The difference belongs to those two algorithms on that component rather than to the on-policy and off-policy families, since SAC is the most accurate of the algorithms that pass their check on the cost-target component and fails its validity check on the inventory problem.

The runs on this axis share the memory study's control problem and grid (Appendix B.2). We subtract an algorithm's informed-condition shortfall from its uninformed-condition shortfall, which removes whatever it loses for reasons unrelated to the latent regime, since the informed problem is Markov in its input. What remains is the penalty it pays for having to infer the regime. With resampling over seeds, feedforward PPO across the full λ range pays 0.13% with bootstrap interval $[0.07, 0.19]$ and recurrent PPO 0.11% with $[0.05, 0.18]$. TD3 pays 0.80% with interval $[0.49, 1.15]$ and exceeds feedforward PPO at the same window by 0.62% with interval $[0.30, 0.99]$. We do not pool the off-policy algorithms against the on-policy ones, since we excluded SAC's runs for failing their validity check and cannot then count them in the statistic that would carry the claim. Off-policy evaluation from data generated in a partially observable environment is biased in general (Tennenholtz et al., 2020). What we measure is consistent with that result and does not demonstrate it.

Bootstrapping is not the mechanism behind the on-policy against off-policy split of Section 5. PPO with GAE, whose parameter λ interpolates between one-step bootstrapping and Monte Carlo returns, is near Monte Carlo, while TD3 and SAC bootstrap one step, so the split is confounded with the λ axis, a concern the λ-discrepancy literature makes precise. The λ-discrepancy of Allen et al. (2024) is zero for a Markov representation and almost always non-zero for a broad class of partially observable ones. We stated a prediction before running, that lowering λ should hurt only the uninformed condition, because the uninformed input is not Markov while the informed one is, and its own control refutes it. Lowering λ from 0.95 to 0 costs the informed condition 0.63 percentage points, from 0.29% to 0.92% above its controller, and costs the uninformed condition 0.48 percentage points, from 0.46% to 0.94%. Bootstrapping with a learned critic costs about half a percentage point here whichever condition it runs in, and the penalty for having to infer is smallest at $\lambda = 0$, the opposite of the prediction, so bootstrapping cannot explain a split that appears only in the uninformed condition. The λ axis varies the length of the backup inside an on-policy algorithm, so it leaves untested the other route the split could run through, the mismatch between a replay buffer's belief distribution and the current policy's, which partial observability sharpens.

F THE INVENTORY RANKING STUDY

In the inventory ranking study, we test whether the gap measured from trained agents agrees with $\mathcal{V}$. We fixed 12 inventory configurations, listed in Appendix B.2, that reach further than the cost-weight component and span $\mathcal{V}$ from 0% to 31.6%. We asked whether the gap measured with PPO ranks them the way $\mathcal{V}$ does, under the hypothesis, fixed in advance, that it does not. It does, at a rank correlation of 0.993 with bootstrap interval $[0.986, 1.000]$ and a Pearson correlation of 0.993, so the measured gap matches the level and not merely the order. We asked, as a secondary hypothesis, for at least one pair the two order oppositely *with non-overlapping intervals*, and there is none among the 66 pairs. One pair is inverted on the point estimates, the two configurations whose $\mathcal{V}$ is 11.8% and 14.3%, which separates 0.993 from a perfect 1.000, and their intervals overlap. At the configuration where $\mathcal{V}$ is identically zero the measured gap was -0.011% with bootstrap interval $[-0.037, 0.021]$, so the zero-value control passed (Figure A3).

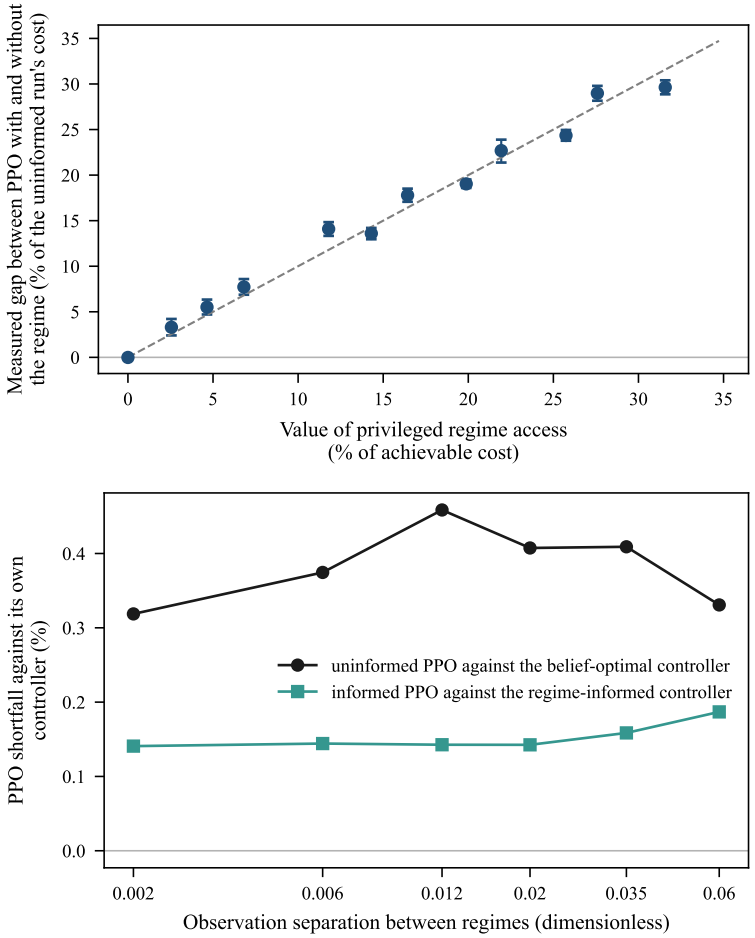


Figure A3: Top, the inventory ranking study: the gap measured between PPO trained with and without the regime one-hot, as a percentage of the uninformed run's own cost, against $\mathcal{V}$, the value of privileged regime access, which is a percentage of achievable cost, across the 12 inventory configurations, with bootstrap intervals and the identity line dashed. Bottom, the identifiability study: PPO's shortfall against its own controller in each condition across the six observation separations, each as a percentage of that controller's cost, uninformed against the belief-optimal controller and informed against the regime-informed controller.

G THE IDENTIFIABILITY STUDY

In the identifiability study, we test whether an agent's shortfall against its own controller grows as the regime becomes harder to identify. We held one control problem fixed and degraded the observation across six levels built for the training runs and listed in Appendix B.2. Its cost of acting differs across regimes, $c = (1, 4)$, so the regime moves a control parameter and this is not the observation component of Section 4, which fixes $c = (2, 2)$. Our hypothesis, fixed in advance, was that the uninformed agent's shortfall against the belief-optimal controller grows as inference hardens. It does not, and the shortfall stays between 0.32% and 0.46% at every level while $\mathcal{V}$ itself spans 1.55% to 7.42%, and its correlation with difficulty is slightly negative (Figure A3). As a secondary check fixed in advance, we demanded a rank correlation of the informed-condition shortfall across the six observation separations, at $|r| < 0.5$, and it reads -0.771. That shortfall runs from 0.14% to 0.19%, so the statistic ranks a series whose whole range is 0.05 percentage points, and a scale-free statistic flags such a flat series as drifting. This is the one threshold that binds. That check's verdict carries no information about the primary hypothesis, and we record both outcomes in Table 1.

The reference point moves with the difficulty, which is what we think leaves the shortfall flat. The belief-optimal controller reads the same observations the agent reads, so as the regime becomes harder to identify that controller also loses ground, and the agent is matched against a target that recedes with it. We score the same runs against the regime-informed controller instead, whose information does not degrade, and the shortfall then rises from 1.91% at the widest observation separation to 8.36% at the narrowest, at a rank correlation of 1.000 with difficulty. Almost the whole of that rise is $\mathcal{V}$ rather than anything the agent does, so what grows as inference hardens is the part no agent restricted to the observations reaches and not the part this agent fails to collect. The prediction we fixed in advance would have been supported against the regime-informed controller, and the choice of reference point is what decides between the two readings.

H THE TARGET-SEPARATION STUDY

In the target-separation study, we test whether the measured gap ranks $\mathcal{V}$ on a problem whose action is unbounded. We run the cost-target component of Section 4 as a scalar tracking problem over five target separations listed in Appendix B.2, with the cost of acting common to both regimes. Where the masked condition reaches its own bound, at the three widest separations, the relative improvement it concedes to an adapting agent runs from 40.4% to 55.8% against an available 41.8% to 56.0%. The masked condition stays within 3.2% of the no-information bound at four of the five separations and 54% above it at the fifth, a training failure at that configuration and not a reading of the bound. Shortfalls in the uninformed condition against the belief-optimal controller run from 1.88% to 5.98% for the two PPO variants, against 0.05% to 6.27% for feedforward PPO on the inventory problem, so the cost-target component is no harder to learn even though its action is unbounded. At zero separation feedforward PPO, recurrent PPO and SAC measure 0.66%, 0.06% and 0.46% at six seeds against an exact zero, and recurrent PPO's verdict is the 30-seed reading of Appendix B.4 (Figure A4).

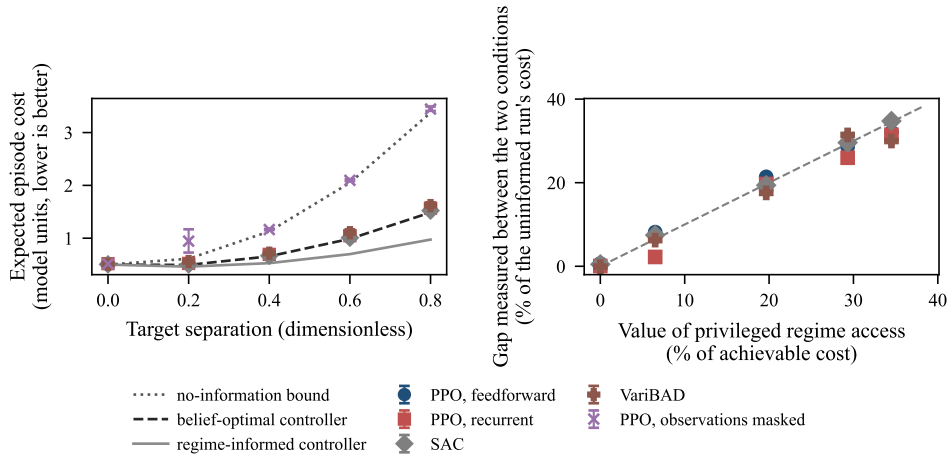


Figure A4: Agents on the cost-target component, a scalar tracking problem in which the latent regime moves the target. Left, expected episode cost in the uninformed condition against the three reference values, the regime-informed controller, the belief-optimal controller and the no-information bound, as the separation between the two regimes' targets widens. Right, the gap measured between the uninformed and informed conditions, as a percentage of the uninformed run's own cost, against $\mathcal{V}$, the value of privileged regime access, which is a percentage of achievable cost, with the identity line dashed. Markers are means over 6 seeds, with percentile intervals on the left panel. The masked condition has its observation window zeroed. TD3, RL^2 given the action only, and the context encoder fail their validity check here and are not plotted. VariBAD is the action-only variant, averaged over 30 seeds for its check and plotted at six like the rest.

Failures concentrate at the smallest non-zero target separation, where $\mathcal{V}$ is smallest: the context encoder's informed run misses its check there, and recurrent PPO's largest measurement error is there.

Every algorithm reads the observation window and the controllable state. The two meta-reinforcement learning algorithms additionally take the previous action, and in their published forms the previous reward as well. On the cost-target component the two regimes share their cost parameters and differ only in the target, so the previous reward carries information about the regime that the observation stream does not, and the belief-optimal controller is a floor for the observation history rather than for a policy that reads it. We therefore run both algorithms with the previous action only, each at the budget its own zero-separation calibration selects, 4,000 iterations for both. RL^2 is then unstable, and its informed run averages 98.5% above its controller at the smallest non-zero separation, where one seed ends 478% above and two others above 26%. Seeds diverge at every other non-zero separation as well, in both conditions. VariBAD reads 16.3% at the widest separation on six seeds and 10.8% at 30, which is the escalation Appendix B.4 required, so it passes the check. Its uninformed runs end 8% to 12% above the belief-optimal controller, and its measured gap follows $\mathcal{V}$ at a mean absolute error

of 1.0 percentage point, 8.9% against 6.5%, 20.7% against 19.7%, 29.1% against 29.3% and 33.1% against 34.5%. With the previous reward as well, at budgets of 1,200 and 2,400 selected the same way, RL^2 passes the check at 7.85%, recovers $\mathcal{V}$ at a mean absolute error of 1.6 percentage points and a rank correlation of 1.000, and fails the zero-value control at -2.5% on an interval that covers zero, while VariBAD misses at 15.2% on six seeds and at 22.7% on 30. Their uninformed runs then stay above the belief-optimal controller at every separation, by 3.9% to 6.2% and 1.4% to 17.7%, which does not contradict the sufficient-statistic argument of Section 3 but is not bounded by it.

The context encoder passes the observation window through a linear layer to the shared hidden width, a tanh, a linear layer to eight dimensions and a second tanh, and the policy and value heads read that summary together with the embedded state. The informed condition bypasses the encoder with the regime one-hot. We chose its training budget the same way as for the other algorithms, at 4,000 policy iterations. At that budget the informed condition is 66.0% above its controller at separation 0.2 against the 15% validity check and the zero-value control reads 4.3% against 1%, both noted in Table 1 and its caption. A shorter budget of 1,200 iterations leaves the informed condition 27% above its controller at the same separation, so the failure is instability rather than underfitting. The failures in this study concentrate at separation 0.2 across architectures.

We reran the four algorithms of this study with the action squashed into a box, to test whether TD3's divergence here is ours rather than its own. The divergence is a mismatch of parameterization on both sides. Bounded, TD3's informed run comes within 2.2% to 8.5% of the regime-informed controller at every separation and passes the check it fails unbounded, while its uninformed run stands 553% to 16,393% above the belief-optimal controller, so the two runs read a failure to infer on a control problem the algorithm has solved. Under the same bound SAC's informed runs stand 856% to 8,181% above their own controller and recurrent PPO misses its check at separation 0.4, at 48.1% against 5.7% unbounded. Feedforward PPO is the only one of the four that passes under both. We report the study unbounded because that is the parameterization the controllers minimize over, and because no single choice admits all four.

I THE CROSS-ENVIRONMENT RANKING

Ranking agents by $\mathcal{I}$ and by s can disagree, and how often depends on which run pairs we admit. We count a run pair only where both orderings resolve, which means the 95% bootstrap interval of the difference excludes zero on both statistics. Over all 181 resolved run pairs the two disagree on 45. Restricting to pairs whose baselines both reached their own bound leaves 18 of 116, restricting further to pairs from the same problem family leaves 8 of 68, and requiring that neither algorithm came in under its own bound leaves 8 of 63.

The last of these is the count we report. The strata above it admit an undertrained baseline, whose ε inflates $\mathcal{I}$ by construction, a comparison between two unrelated problem families where a difference in $\mathcal{I}$ is unsurprising, and algorithms whose measured cost fell below a bound no feasible policy can beat. All eight surviving inversions are on the cost-target component, and each pits an algorithm at a narrower target separation against one at a wider separation, where the narrower run collects the larger fraction of what was available and reports the smaller $\mathcal{I}$.

J THE CALIBRATED-AGENT STUDY

In the calibrated-agent study, we test whether the controllers predict what a trained agent earns once the environment's parameters come from data rather than from a knob. We fixed the selection rule before the corresponding runs, and it reads no value or gap column when it picks its configurations: ten stocks, one per decile of dollar volume, each the stock whose fitted volatility ratio is closest to its decile's median. The problem is liquidation under linear temporary impact over a 78-step horizon, with risk aversion set by the bisection on first-step urgency the calibration fixes, at the conservative end of its band, where the first step in the low-volatility regime is 4 times the constant-rate schedule's. Their $\mathcal{V}$ runs from 0.04% to 22.1%, and the two smallest serve as a control. We recompute the controllers under the agent protocol, since the calibration used the protocol for a comparison with no agent in it.

The informed runs of feedforward PPO, recurrent PPO and SAC finish 105%, 602% and 229% above their regime-informed controllers at their worst configuration, and no algorithm passes on any of the ten. The masked runs finish 282% above the no-information bound, so no algorithm reaches either reference. Under the rule fixed in advance no algorithm is admitted and the hypotheses are not read.

The failure is uniform rather than concentrated, so it is not a property of one configuration. Two things separate these environments from the constructed ones where the same algorithms reach within 2% to 11% of their controllers. The horizon is 78 steps against 26, and the budget is the one the selection rule chose at that horizon. We vary each in turn, the shorter horizon at the same budget and the same horizon at four times it. The configurations, the selection rule and the thresholds stayed fixed, and we recomputed the controllers at whichever horizon the cell uses.

At 26 steps on the same calibrated parameters and the same budget, SAC comes within 9.6% of its controller on every one of the ten configurations and passes, from 229% at 78 steps, and the two on-policy algorithms improve to 30% and 55% from 602% and 105%, so the horizon rather than the optimization budget is the binding constraint. The masked runs fall from 282% above the no-information bound to 59% at their worst. We raise the budget instead, and the study gets worse. At 78 steps and four times every algorithm's budget, feedforward PPO's median informed shortfall goes from 52% to 3,069% and its uninformed runs from 33% to 5,805%, while SAC, which is off-policy, improves from 229% to 83% and still fails. Training longer at this horizon diverges for the on-policy algorithms rather than converging, so the algorithms are unstable at 78 steps rather than underfitted, and the same fourfold increase that destroys them helps the one algorithm with a replay buffer. The protocol holds the learning rate fixed across horizons at 3×10^{-4} for every algorithm, which is the continuous-control default Schulman et al. (2017) report for PPO, so that instability is confounded with a step size no run at this horizon ever adjusted. We vary the horizon and the budget and never the step size, so another step size may train at 78 steps.

We therefore report the study as inconclusive on its own hypotheses and informative about where the instrument reaches. The reference values are computable at 78 steps, and the difficulty is in training an agent there. We also name two hedging problems at 10 and 21 days in this design, and we built and checked them against the training code but did not train on them. Under the rule we fixed in advance we select a budget at the configuration where the regime carries no information, and for these two we found no budget among the four we tried. A liquidation episode costs about 1 in the calibration's units and a hedging episode about 0.0003, more than three orders of magnitude apart, so we also tried rescaling every cost by its own no-information bound. That rescaling is a change of units, which leaves the optimal policy and both measured quantities exactly where they are, and it left the hedging problems as unreadable as before while making liquidation worse, so we reverted it. We report the two as unreadable at the budgets we admit here rather than train them at a budget that rule would have refused.

Table A3: The accounting on four POBAX domains, from our own runs of its released code at commit `a5e1d62` under the hyperparameters its configurations select. R is the mean return over completed episodes across 30 seeds. $\mathcal{G} = R_{\text{reference}} - R_{\text{baseline}}$ is the gain the domain makes available, in the units of R rather than the fraction Equation (3) defines, and it appears under each domain's name. s is the fraction of $\mathcal{G}$ an algorithm collects, with a 90% interval from resampling the seeds of the algorithm and the baseline, and of the reference too where it is a trained run, so an interval can reach above 100%. $\mathcal{I}$ is the improvement over the baseline as a fraction of the baseline. The baseline is what s and $\mathcal{I}$ are measured from, so its own cells are blank. The T-maze and Battleship references are the analytic optima, 4.0 and 87, and the RockSample references are the trained full-state runs. Battleship has no Transformer-XL row because the released implementation errors out on that observation space.

Domain	Algorithm	R	s	90% interval	$\mathcal{I}$
T-maze 10	baseline	1.604			
$\mathcal{G} = 2.396$	recurrent PPO	3.709	87.9%	[75.9, 97.2]	131.2%
	λ-discrepancy	3.632	84.6%	[69.0, 95.9]	126.4%
	Transformer-XL	4.000	100.0%	[100.0, 100.0]	149.4%
RockSample(11,11)	baseline	9.975			
$\mathcal{G} = 7.880$	recurrent PPO	11.202	15.6%	[6.6, 38.5]	12.3%
	λ-discrepancy	12.737	35.1%	[18.4, 82.1]	27.7%
	Transformer-XL	17.311	93.1%	[52.2, 217.3]	73.5%
RockSample(15,15)	baseline	10.002			
$\mathcal{G} = 12.703$	recurrent PPO	10.646	5.1%	[1.5, 13.8]	6.4%
	λ-discrepancy	17.858	61.8%	[36.2, 145.7]	78.5%
	Transformer-XL	21.356	89.4%	[54.1, 205.2]	113.5%
Battleship	baseline	12.482			
$\mathcal{G} = 74.518$	recurrent PPO	12.468	-0.019%	$[-0.7, 0.7]$	-0.11%
	λ-discrepancy	16.664	5.6%	[4.6, 6.7]	33.5%

K THE POBAX ACCOUNTING

POBAX (Tao et al., 2025) screens its domains for the memory improvability we describe in Section 2, and that design supplies all three of our information sets without any modeling on our part. Observations without memory give the non-adapting baseline, the full state without memory gives the full-state reference, and the memory-equipped runs are the third set, which we fill with recurrent PPO, the λ-discrepancy algorithm of Allen et al. (2024) and Transformer-XL (Parisotto et al., 2020). We take each cell at the hyperparameters its released configurations select and run 30 seeds, which is POBAX's own protocol, so we select nothing. We pin its repository at commit `a5e1d62` and release the command behind every cell, the per-seed score behind every cell, and the raw T-maze evaluation arrays we use in the next paragraph, so the accounting below runs without the cluster the training ran on.

We score a run as the mean return over completed episodes in the final evaluation. The field the suite logs carries the last completed episode's return forward across timesteps. `parse_batch_experiments.py` in the released code sums that field over time and divides by the number of completions, and that reduction multiplies every condition by its own average episode length. On the T-maze it returns 33.3, 114.0 and 48.3 for the baseline, recurrent PPO and the full-state reference, so it puts an algorithm above its own ceiling. Its figures plot online training curves from a separate record, and the equivalent block in `parse_no_frame_experiments.py` is commented out.

Two domains carry a ceiling that needs no training, and both follow from the environment definitions at the pinned commit. `envs/jax/tmaze.py` sets `good_reward = 4.`, paid once per episode, so 4.0 is the optimum. `envs/jax/battleship.py` sets `reward = (rows*cols)*done - (1-done)`, so a return is 101 minus the number of turns, and `envs/__init__.py` sets `ship_lengths = (5, 4, 3, 2)`, so an agent that sees the board needs $5 + 4 + 3 + 2 = 14$ turns and scores 87. We report both against those values in Table A3. Against the trained full-state reference instead, the T-maze figures are 90.1%, 86.8% and 102.6%, and the last of these exceeds the reference because Transformer-XL reaches the optimum while the reference run reaches only 3.939.

A 100 point terminal reward sets the origin of Battleship's return scale where nothing in the problem sets it, so dividing a gain by the baseline return measures partly that choice. The contrast between a reported 33.5% and a collected 5.6% is the sharpest in Table A3 and the least well grounded, and the quantity to read there is the fraction collected, which divides by the distance to a ceiling the problem does fix at 14 turns. An additive shift of the reward origin cancels in that fraction, since it enters the algorithm's gain and the reference's gain alike, and it does not cancel in the improvement reported, which divides by the shifted baseline.

The ratio $\mathcal{G}/|R_{\mathrm{baseline}}|$, which takes the place of $\mathcal{A}$ on a return scale, is 1.49, 0.79, 1.27 and 5.97 across the four domains, so $\mathcal{I}$ runs above s in three of them and below it in the fourth. Within one domain the two rankings cannot differ, since both divide an algorithm's gain over a common baseline by a constant that does not depend on the algorithm, so only an aggregate over domains whose orderings disagree could separate them. Ranking the three algorithms by the mean of either quantity gives Transformer-XL, then the λ-discrepancy algorithm, then recurrent PPO. Recurrent PPO beats the λ-discrepancy algorithm on the T-maze and loses to it on both RockSamples, so weights that lean on the T-maze order those two one way under s and the other way under $\mathcal{I}$, and equal weights do not. Transformer-XL is first under any weights, since it wins every domain it runs on.

Transformer-XL has no Battleship entry. The released implementation reads the shape of the observation space, and Battleship returns a dictionary space, so the run errors out before training. POBAX's released configurations still select hyperparameters for that cell, so the suite tunes a run it cannot execute. We report the domain with the two algorithms that run on it.

L Using the Accounting on Another Problem

We give here what a reader needs to run each part of the accounting on a problem of their own, and what each part costs, since the three parts ask for very different things.

The bound asks for nothing beyond a number already in print. Feedforward PPO in the target-separation study reports a 45.8% improvement at a target separation of 0.2 and 40.6% at 0.4, so the reported number ranks the first configuration above the second. The environment offered 19.3% and 41.8%, and the masked baseline stood 54.4% and 3.1% above its own bound, so by Equation (4) the algorithm collected 84.5% of what was available in the first and 92.7% in the second. The reported ordering is the reverse of the collected one, and the whole of the difference is the baseline rather than the algorithm. The first of those two configurations is one of the four we set aside in Section 6 for an undertrained baseline, which is what that screen is for.

Reading the bound on a published number needs a ceiling on how far the baseline falls short, and our own runs argue against assuming that ceiling is small. Across those four configurations ε runs from 2.1% to 54.4%, a factor of twenty-six within one control problem and one algorithm. Assuming a well-trained baseline assumes the term that decides whether the reported number can be read at all, so we recommend reporting the bound as a band over a stated range of $\bar{\varepsilon}$ rather than at a single value.

The two-run estimate asks for one retraining and no model. We train one algorithm twice on the same environment, once with a regime label appended to the observation and once without it, and the gap between the two evaluation costs estimates what that label was worth there. The label need not be the true regime, and it need not be correct, since it only has to be something the agent can condition on in one run and not in the other. The same pair of runs carries the diagnosis of Section 6. An informed run that misses its own reference is a control failure, and no memory architecture repairs it.

Measuring the fraction rather than bounding it asks for our environments, where the three reference values are known. We supply each configuration through `make_env`, which takes a configuration name and a condition, and its three reference values through `reference_values`, so the training run sits between the two calls. The fraction collected is the agent's improvement over the uninformed cost divided by the belief-optimal controller's improvement over that same cost, which is the quantity we normalize in Equation (3) and the only one of the three that transfers across environments.

M ADDITIONAL EXHIBITS

We collect here the figures the main text points to without showing: the component grid behind Section 4, the transfer check outside the linear-quadratic class, the full decomposition behind Section 6, and the memory study's learner comparisons. In the latent-mode process of Section 4, the composed total differs across control settings, from 1.0% to 46.6%, while it stays invariant to signal accuracy within each. We compute $\mathcal{V}$ and $\mathcal{A}$ from values rather than costs there, as the regime-informed value divided by the belief-optimal one and the belief-optimal value divided by the no-information one, each less one, which is the same construction read against a reward scale. The range of $[0, 1)$ that Section 3 takes from the ordering of the three costs belongs to that scale and does not carry to a reward scale, where such a fraction can exceed one, as $\mathcal{G}/|R_{\text{baseline}}|$ does on Battleship in Table A3. The λ-discrepancy of Allen et al. (2024) reads the agent's own value estimates instead, and it peaks in the middle of the accuracy range, where a signal leaves the most for memory to do (Figure A6).

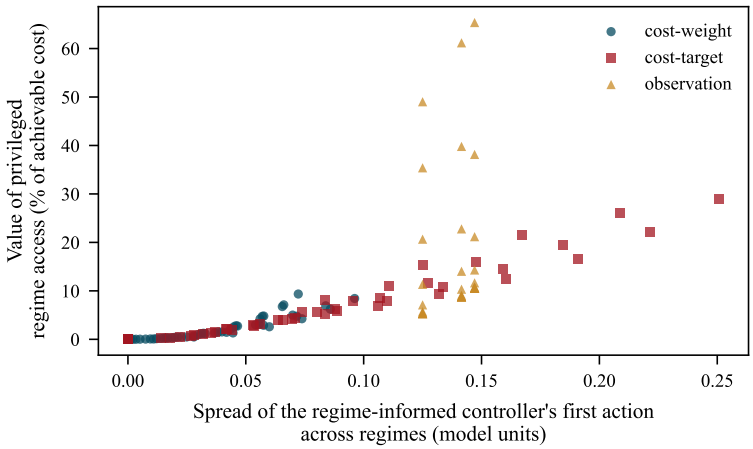


Figure A5: Value of privileged regime access against the spread of the regime-informed controller's first action across regimes, the standard deviation weighted by how often each regime occurs, across the 134 configurations of Appendix B.2 in which we vary one component at a time. Cost-weight varies the cost ratio and the tracking-weight ratio, cost-target the target separation and the cost level, and observation the observation separation and the regime persistence, with each regime's control parameters held fixed across the grid. These grids are not crossed, so they order configurations within a component and cannot compare across them, which Figure 1 does.

In the memory study, we run three PPO variants at four observation separations on the inventory-style control problem, two feedforward with different window lengths and one recurrent, with a GRU. Each reaches the reference point its information entitles it to, the recurrent one within the belief-optimal controller's own Monte Carlo band at every separation (Appendix B.3), and on the same configurations TD3 passes the validity check where SAC does not and averages 1.41% above its controller against at most 0.46% for any PPO variant (Figures A8 and A9).

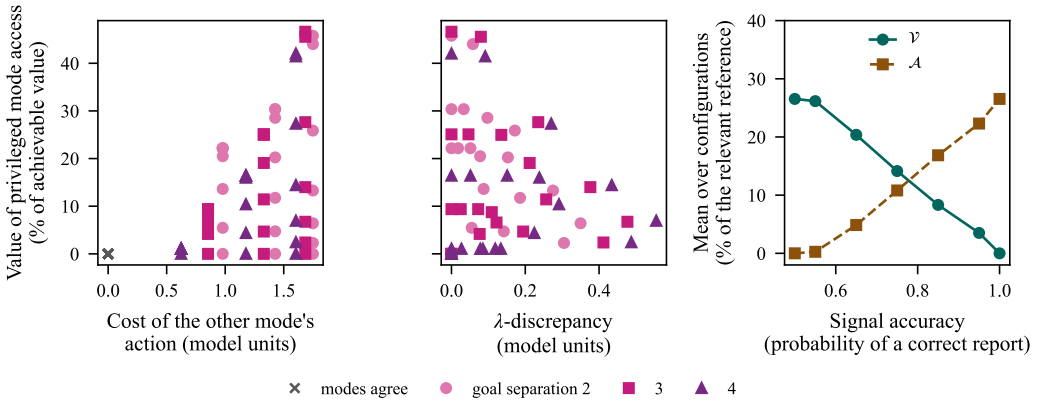


Figure A6: A discrete latent-mode process outside the linear-quadratic class: five positions, three actions, an indicator reward, and a grid of 84 configurations over goal separation, mode persistence and signal accuracy, with all three reference values computed by value iteration. Left and center, two diagnostics against $\mathcal{V}$, where the regime is the latent mode: the cost of taking the other mode's action, which is the displacement diagnostic lifted to discrete actions, and the λ-discrepancy of Allen et al. (2024). Gray crosses mark the configurations where the two modes' goals coincide, so $\mathcal{V}$ is zero. Right, the configurations where the two modes' goals differ, averaged at each signal accuracy, with $\mathcal{V}$ measured against the belief-optimal value and $\mathcal{A}$ against the no-information bound.

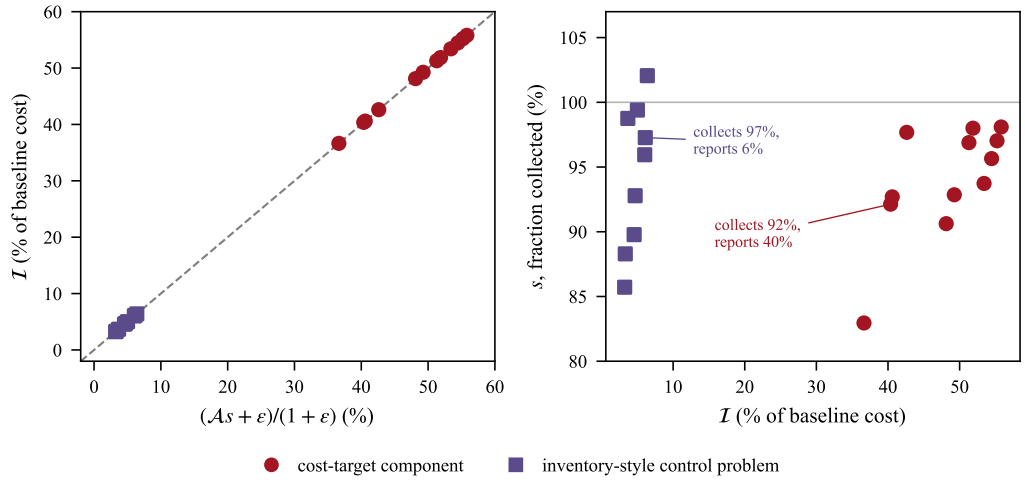


Figure A7: The decomposition of Proposition 3.1 on the 21 trained agent-environment pairs where more than 1% was available and the baseline reached its own bound. Left, $\mathcal{I}$ against $(\mathcal{A}\,s+\varepsilon)/(1+\varepsilon)$ with the identity line dashed. Right, s against $\mathcal{I}$. Circles are the cost-target component and squares the inventory-style control problem, and the two annotated runs are feedforward PPO on that problem and recurrent PPO on the cost-target component. The gray horizontal line marks $s = 100\%$.

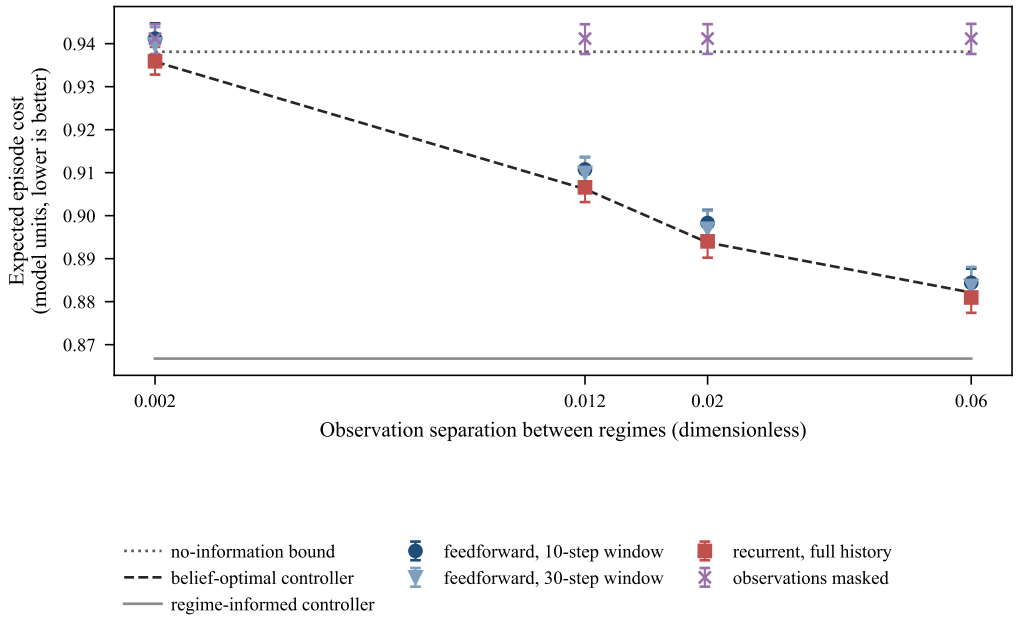


Figure A8: Expected episode cost of three PPO variants in the uninformed condition, feedforward over a 10-step window, feedforward over a 30-step window and recurrent over the full history, together with the 10-step feedforward variant in the masked condition, which has its observation window zeroed, against the regime-informed controller, the belief-optimal controller, and the no-information bound, across four observation separations, with the control problem fixed. Markers are means over 6 seeds with percentile intervals.

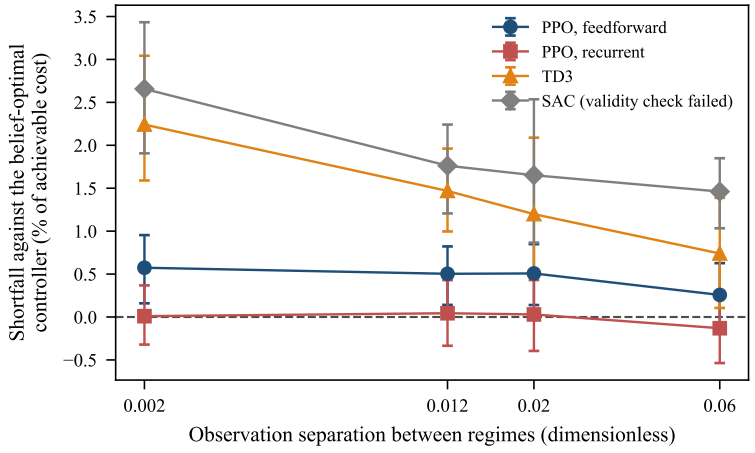


Figure A9: Shortfall of feedforward PPO, recurrent PPO, TD3 and SAC in the uninformed condition against the belief-optimal controller, as a percentage of the achievable cost, across the four observation separations. Inference is harder to the left. Markers are means over 6 seeds with percentile intervals. SAC fails its validity check, is plotted anyway, and is marked as such in the legend.