ACTION-FREE CAUSAL STATE INFERENCE FOR REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

In partially observable environments, the state an agent learns from its history decides how well and how efficiently reinforcement learning succeeds, and causality gives that state its right target, the *causal state*, which groups histories by the future they lead to and helps the agent tell apart precisely the situations that matter, whereas predictive states may also keep irrelevant history. In this paper, we extend causal state learning to the action-free setting, as humans infer the plot of a film without acting in it, and prove that, under a mild condition on data collection, the causal state learned from observations and rewards alone loses nothing relevant to predicting the future and can be even more compact than its action-based counterpart. We further characterize how fast forgetting removes the rest, so its strength can be set automatically. Building on this, Action-free Causal State Inference (ACSI) injects noise into its recurrent memory, so that prediction preserves what the future requires while the rest decays, with the noise self-calibrated to a target forgetting rate rather than tuned per task. Empirically, ACSI recovers ground-truth causal states 9 AUC points better than action-conditioned, contrastive and bisimulation baselines, and is the only frozen representation to solve both control tasks that provably require memory, using 2.6–5.7$\times$ fewer frames than recurrent reinforcement learning. On real robot laser streams, it identifies revisited places 4 AUC points better than all learned baselines, offering a principled route to compact states from action-free data that make reinforcement learning more sample-efficient. The code of our method and environments will be released upon acceptance.

1 INTRODUCTION

An agent that cannot observe its environment's state must act on its history, yet histories grow exponentially with their length while the situations that matter do not, some 10^{30} histories of length 50 arising from only 195 distinct situations in the grid worlds we study. Since the experience an agent needs grows with the number of situations it tells apart (Gheshlaghi Azar et al., 2013), its state should distinguish exactly these situations and nothing more.

Computational mechanics names this state. The *causal state* of a history is its equivalence class under the distribution of futures it induces (Crutchfield & Young, 1989; Shalizi & Crutchfield, 2001), *sufficient* because it preserves every predictable feature of the future and *minimal* because no coarser statistic is sufficient. We learn it without actions, as humans infer the plot of a film without acting in it, from a stream of observation and reward pairs, which we call *words*, and show that under a mild condition on data collection this loses nothing relevant to the future. Sufficiency and minimality remain two requirements, and they are met by different mechanisms.

Existing representation learning meets the first requirement and leaves the second unaddressed, since world models, contrastive prediction and self-predictive objectives train a state to predict the future, which makes it sufficient but places no pressure on what else it retains. The residue is measurable, since inside one minimal state of our relay task a recurrent predictor still encodes how long ago that state was entered, and removing that timing signal leaves 286 effective states against the 279 that task actually has (Table 5). Objectives that do compress define what to keep through actions and rewards, which passive logs do not record, while a rate penalty compresses without saying what to discard (Appendix A).

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

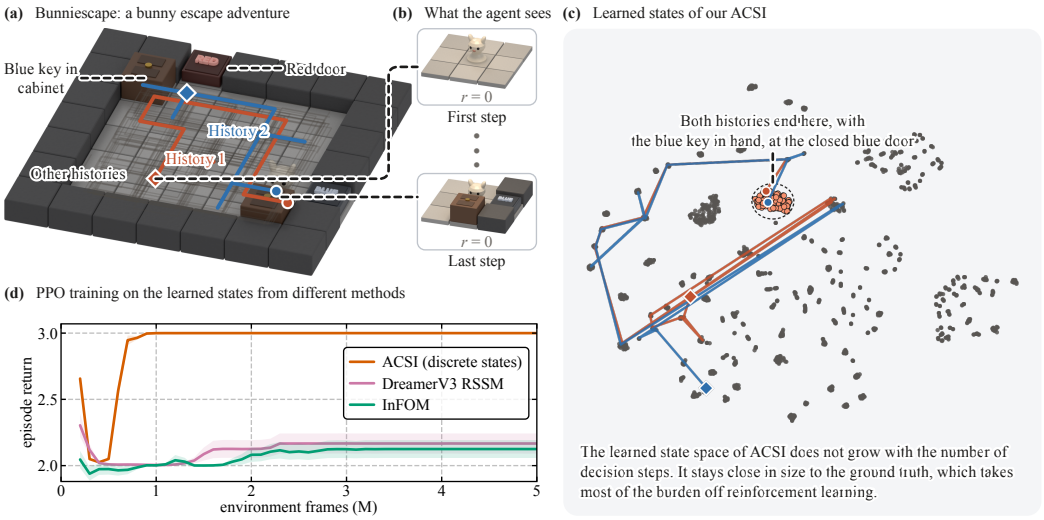


Figure 1: The causal state ACSI estimates and its benefits for downstream reinforcement learning. (a) Two of the many histories the environment generates reach the same causal state. (b) The agent sees only words, an observation with the reward that came with it, which never show the key it carries. (c) The learned state space is about the size of the true one and the two histories end on one state. (d) PPO on the learned state learns faster than on strong and recent baselines.

We propose Action-free Causal State Inference (ACSI), which learns causal states from words alone by training a recurrent encoder to predict the discounted occupancy of future words, making its state sufficient, while noise in its memory erases everything prediction does not restore, making it minimal, so that a policy trained on it spends experience only on the situations that matter. This Gaussian noise limits how much the memory can hold without specifying what to discard, leaving that choice to prediction, and it is the multiplier of a constraint on how strongly one update shrinks the difference between two memories, so we fix a target for that contraction and let a feedback rule move the multiplier.

- **Action-free identification.** We prove that, under a mild condition on the data policy, the causal state learned from words alone loses nothing relevant to the future compared with its action-conditioned counterpart, and can become strictly more compact when an action symmetry is invisible in the words (Propositions 1 and 2).
- **Guarantees for sufficiency and minimality.** We establish that a recurrent encoder predicting the discounted occupancy of future words recovers exactly the causal-state partition, and prove that recurrent noise erases the remaining redundancy at the contraction rate of the recurrence, the noise level acting as a Lagrange multiplier so that one contraction target replaces per-task tuning (Propositions 3, 4 and 6 and Theorem 1).
- **A theory-guided method and its evidence.** Grounded in these guarantees, we design ACSI. It recovers the true states 9 AUC points above the best baseline, is the only frozen state that solves both tasks where control provably needs memory, needing at least 2.6 times and 5.7 times fewer frames than recurrent PPO, and ranks revisits on real laser streams 4 AUC points above every learned baseline, while direct measurements confirm each theoretical prediction (Section 4).

2 Formulation and Theory of Action-free Causal States

An agent interacts with an environment of finitely many latent states and actions under a behaviour policy π, and we never record the actions. What we record at step t is a *word* $w_t \in \mathcal{W}$ pairing an observation with its reward, $w_t = (o_t, r_t)$, so that a statistic sufficient for the future is sufficient for value, and which reduces to the observation alone where no reward is recorded, as in the laser logs of Section 4. We write $H_t = w_{1:t}$ for the history and $F_t = w_{t+1:\infty}$ for the future.

Two histories are *observationally equivalent*, written $H \sim_{\mathrm{O}} H'$, when $P_\pi(F \mid H) = P_\pi(F \mid H')$, and the *causal state* $\sigma(H)$ is the equivalence class of H under this relation (Crutchfield & Young, 1989; Shalizi & Crutchfield, 2001). Action-conditioned methods target a finer object, the interventional class $\iota(H)$, which sends every finite action sequence $a_{1:k}$ to $P(w'_{1:k} \mid H, \mathrm{do}(a_{1:k}))$ with the unrecorded past actions averaged out under π (Appendix B), and histories whose maps agree are *interventionally equivalent*, $H \sim_{\mathrm{I}} H'$, which is the analogue on word sequences of the action-conditioned state of the Causal-state RNN (Zhang et al., 2019) and of bisimulation (Ferns et al., 2004).

A statistic h of the history is *sufficient* when $P_\pi(F \mid H)$ is a function of $h(H)$, and among sufficient statistics it is *minimal* when $I(h; H)$ is smallest, a property the next decomposition makes checkable.

Lemma 1 (Redundancy, classical, Shalizi & Crutchfield 2001; Creutzig et al. 2009)**.** *For any statistic h of the history, $I(h; H) = I(h; \sigma) + I(h; H \mid \sigma)$, and a sufficient h is minimal if and only if the redundancy $I(h; H \mid \sigma)$ vanishes, which happens exactly when h depends on the history only through σ.*

We want a recurrent statistic $h_t = f_\theta(h_{t-1}, w_t)$ that is sufficient and of small redundancy, since the experience a learner needs grows with the number of situations its state tells apart (Gheshlaghi Azar et al., 2013), and Lemma 1 shows that one objective cannot serve both ends, since a predictive loss constrains $I(h; \sigma)$ alone and leaves the redundancy large (Figure 1), so the rest of this section closes the gap in four steps, from dropping actions through sufficiency and minimality to the strength of the mechanism that provides it.

Learning without actions. Words alone lose nothing an action-conditioned method would keep, provided the data policy itself looks no deeper than the interventional class.

Assumption 1 (Behaviour policy)**.** The policy depends on the history only through the interventional class, $\pi(a \mid H) = \tilde{\pi}(a \mid \iota(H))$. The uniform random policy and any Markov policy on ι satisfy this.

Proposition 1 (Ignoring actions loses nothing about the future under π)**.** *Under Assumption 1, $H \sim_{\mathrm{I}} H'$ implies $H \sim_{\mathrm{O}} H'$. The causal state is therefore never finer than the interventional class, of which it is a coarsening, and it is sufficient for predicting the future under π.*

The guarantee is sufficiency under the sampling distribution, hence for evaluating π. The causal state can moreover be strictly coarser than the interventional class, which is where it saves states and also where it may drop what another policy needs.

Proposition 2 (Invisible action symmetries shrink the state)**.** *Let π be uniform and let a group G act on latent states and actions such that transitions are equivariant, emitted words are invariant and the initial distribution is G-invariant. Then causal states are unions of G-orbits, and whenever some action sequence distinguishes a state s from $g \cdot s$, the interventional partition is strictly finer.*

The guarantee therefore extends to control except under such a symmetry (tested in Section 4) or a collector whose own routine violates Assumption 1 and enlarges the state (Proposition 5), with proofs in Appendix B, so actions can be dropped and the question becomes which signal recovers this state from words.

Sufficiency through occupancy. Exact next-word prediction is such a signal (classical, Shalizi & Crutchfield, 2001) but is a poor training signal here, since almost every word is uninformative and the events that matter lie tens of steps away, so we predict instead one quantity that weighs the entire future at once, the γ-discounted occupancy of future words,

$$m_\gamma(H) = \sum_{k \geq 1} (1-\gamma)\gamma^{k-1} P_\pi(w_{t+k} = \cdot \mid H) \in \Delta(\mathcal{W}). \tag{1}$$

Fitting m_γ is enough only if its level sets are the causal states, which holds for all but finitely many γ when distinct causal states differ in the marginal distribution of some single future word, and a computable certificate, positive in all our tasks for every $\gamma \geq 0.9$, verifies it at a given γ (Proposition 6). A recurrent encoder needs no such condition, because histories mapped to the same state predict the same occupancy after every continuation, and this closure alone recovers the causal states (a separating example is in Appendix B.4).

Proposition 3 (Update-closed occupancy equivalence)**.** *Write $H \approx_\gamma H'$ if for every finite word sequence u either u has probability zero after both histories, or it has positive probability after both and $m_\gamma(Hu) = m_\gamma(H'u)$. On a finite causal-state process, $\approx_\gamma$ is the causal-state partition for all but finitely many $\gamma \in (0, 1)$, with no further condition.*

Minimality through contraction. Predicting m_γ thus secures sufficiency, but by Lemma 1 its loss is blind to redundancy, so minimality needs a separate mechanism, noise in the recurrent channel. A bounded state carrying Gaussian noise is a Gaussian channel, whose capacity caps the redundancy of any sufficient h at $C - H(\sigma) + H(\sigma \mid h)$ with $C = \frac{d}{2}\log(1 + B^2/\sigma^2)$ (classical, Cover & Thomas, 2006). The state must be bounded for that cap to bind (Section 4), and a single step is far too loose a channel to explain what we measure (Appendix B.5). Noise accumulated over many steps is what removes redundancy, as the following theorem quantifies.

Theorem 1 (Multi-step forgetting)**.** *Let $h_k = f(h_{k-1}, w_k) + \varepsilon_k$ with $\varepsilon_k \sim \mathcal{N}(0, \sigma^2 I)$ independent of the environment, let two histories read the same words after step s under one coupled noise sequence, and write $\mu_t = h_t - \varepsilon_t$, $\mu'_t = h'_t - \varepsilon_t$.* (i) *For every $t > s$, with no assumption on f,*

$$\mathrm{KL}\big(P(h_t \mid H, w_{s+1:t}) \,\big\|\, P(h_t \mid H', w_{s+1:t})\big) \le \frac{1}{2\sigma^2}\,\mathbb{E}\|\mu_t - \mu'_t\|^2, \tag{2}$$

and any X fixed before step s, such as the start position, obeys the same bound on $I(X; h_t \mid w_{s+1:t})$, with a supremum over pairs of values of X on the right. (ii) *Suppose that $\|F_K^c(h) - F_K^c(h')\| \le \rho_K\|h - h'\|$ with $\rho_K < 1$ for all h, h', where F_K^c composes $h \mapsto f(h, w_k) + c_k$ over any block of K words with any offsets c_k. Then $\mathbb{E}\|\mu_{s+mK} - \mu'_{s+mK}\|^2 \le \rho_K^{2m}\mathbb{E}\|h_s - h'_s\|^2$ and $I(X; h_{s+mK} \mid w_{s+1:s+mK}) \le \rho_K^{2m}\mathrm{diam}^2/(2\sigma^2)$ at every block boundary $m \ge 1$.*

Part (i) is the channel and part (ii) the pace of forgetting, read off as the residual $\hat\rho_K = \|\Delta h_{t_0+K}\|/\|\delta\|$ a perturbation δ leaves after replaying the same words for K steps, while contraction alone forgets nothing, since the bound diverges as $\sigma \to 0$, and causal information survives, since different causal states are predicted differently and occupy basins between which the recursion does not contract. Contraction is not a property a trained predictor has on its own, since the residual of a prediction-only recurrence plateaus rather than decaying geometrically, so the injected noise supplies both the erasure and the hypothesis of part (ii) (Section 4), and the conditioning on the shared words is discussed in Appendix B.

Calibrating the noise. Forgetting now has a mechanism and a pace, and what Theorem 1 leaves open is the strength σ itself, which the next result identifies.

Proposition 4 (Noise is a contraction multiplier)**.** *To second order in σ, training with recurrent noise minimizes $\mathcal{L} + \frac{\sigma^2}{2}R$, in which $R = \sum_t \sum_{s \le t} \|G_t^{1/2} J_{t,s}\|_F^2$ measures how much a perturbation of the past leaks into later predictions, with $J_{t,s}$ the state Jacobian and G_t the Gauss–Newton matrix of the loss (classical, Bishop, 1995; Lim et al., 2021). For the problem of minimizing $\mathcal{L}$ subject to $R \le \rho^{\star 2}$, the quantity $\sigma^2/2$ is therefore the Lagrange multiplier, and exponentiated dual ascent moves $\log\sigma$ by the constraint violation.*

Since R cannot be measured, we control the median residual that can, at the horizon $K = 1/(1-\gamma)$ of the prediction target,

$$\sigma \;\leftarrow\; \sigma \cdot \exp\big(\eta\,(\hat\rho_K - \rho^\star)\big). \tag{3}$$

One target $\rho^\star$ therefore transfers across tasks that share the horizon K while no single σ does, because what is held fixed is the constraint while its multiplier can differ by orders of magnitude (Section 4). What Equation (3) does and does not inherit from Proposition 4, why the prediction loss cannot set σ instead, and the feasible range of the target given by Corollary 1 are in Appendix B.7.

Remark 1. The controller of Equation (3) holds $\hat\rho_K$ at a target $\rho^\star < 1$, a median over perturbation directions where part (ii) asks for a supremum over pairs of states, so it is a proxy for that hypothesis rather than a verification of it (Appendix F).

3 Action-free Causal State Inference

ACSI turns the four results of Section 2 into one pipeline (Figure 2), in which a bounded recurrent encoder reads words alone, a linear head predicts their discounted occupancy, a second training phase

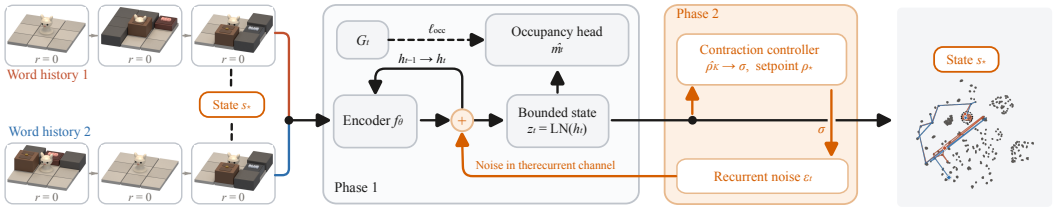


Figure 2: Action-free Causal State Inference in one pass. Words enter a bounded recurrent state that a linear head reads for the occupancy target, and phase two carries that state through noise in the recurrent channel while a controller holds the contraction at the setpoint $\rho^\star$ (Equations (4) to (7)). Both histories of the environment of Figure 1 end in the one embedded state that carries the same tag.

injects noise into the recurrent channel, and a controller sets that noise by holding the contraction at a target, after which a code book can turn the state into discrete symbols. We build it in that order, training the encoder and its target in phase 1, adding the noise and its controller in phase 2, and reading the frozen state out at the end, each component adding one guarantee to the estimate and each checked by an experiment of Section 4. What is new relative to noisy recurrent training (Bishop, 1995; Lim et al., 2021) and to variational bottlenecks is not the noise but its placement and calibration (Algorithm 1 in Appendix C).

Phase 1: a bounded recurrent state that reads words. The encoder consumes one word per step and nothing else, which is all Proposition 1 needs for the state to lose nothing an action-conditioned method would keep,

$$h_t = f_\theta(h_{t-1}, Ew_t), \qquad z_t = \mathrm{LN}(h_t), \tag{4}$$

with f_θ a single-layer GRU (Cho et al., 2014) of width $d = 128$ and E a word embedding. The recursion is deterministic, which makes Proposition 3 apply, since two histories carried to the same state keep predicting alike. The layer norm keeps the state bounded, without which the Gaussian-channel bound of Section 2 caps nothing and noise constrains nothing (Appendix B.5).

Phase 1: sufficiency through the occupancy target. Next-word prediction is an ineffective training signal here, because nearly every word is uninformative and the events that determine value lie tens of steps ahead, so a linear head $\hat{m}_t = \mathrm{softmax}(Wz_t)$ is fitted to m_γ of Equation (1) at $\gamma = 0.95$ under the cross entropy

$$\ell_{\mathrm{occ}} = -\sum_t G_t^\top \log \hat{m}_t, \qquad G_t = (1-\gamma)\,\mathbf{1}[w_{t+1}] + \gamma\big[(1-\lambda)\,\mathrm{sg}(\hat{m}_{t+1}) + \lambda\, G_{t+1}\big], \tag{5}$$

in which sg is the stop-gradient, so that one target carries the whole future out to the horizon $K = 1/(1-\gamma) = 20$ steps. The bootstrap in Equation (5) follows successor-representation learning (Dayan, 1993) with TD(λ) (Sutton, 1988), and since bootstrapping from an untrained head would destroy the long memory before it forms, λ decreases from 1 to 0.8 over the first 200 updates, which raises the state-recovery AUC in one room (Section 4). The state is now sufficient by Proposition 3, and this encoder alone already recovers the true states at an AUC of 0.958 (Section 4), but nothing yet stops it from carrying redundancy.

Phase 2: minimality through noise in the recurrent channel. Lemma 1 removes redundancy among sufficient statistics rather than among all, and noise applied before one exists destroys memory the encoder has not finished forming, so the constraint is switched on only in a second phase. Phase 1 fits Equation (5) without noise until early stopping, and phase 2 restarts from its best parameters and perturbs the state the cell has just produced,

$$h_t = f_\theta(h_{t-1}, Ew_t) + \varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}(0, \sigma^2 I), \tag{6}$$

this same noisy h_t reaching LN and the next step alike, which is the channel Theorem 1 speaks about and the only place noise enters. With the bounded state of Equation (4), Equation (6) is a Gaussian channel whose capacity caps what the recursion may carry (Cover & Thomas, 2006). The placement matters, since noise on the input embedding, or on z_t alone with none fed back, crosses no bottleneck history must pass and leaves the within-state spread unchanged, while the recurrent

noise also supplies the contraction that Theorem 1 assumes (Section 4). Inference and validation are noise-free, because Equation (6) constrains what training may transmit rather than the deployed encoder, and checkpoint selection is suspended for the first 300 updates of phase 2, when the noise raises the held-out loss before merging anything. The state now forgets what the future does not need, its within-state spread falling from 0.159 to 0.138 at no cost in AUC (Section 4), and a one-step consistency plug-in carries the same channel to a causal transformer, which has none of its own (Section 4).

Phase 2: a controller on contraction rather than a tuned noise level. Proposition 4 identifies $\sigma^2/2$ as the multiplier of a contraction constraint, and a multiplier is not a quantity that can be held fixed across tasks, so we fix the constraint $\hat{\rho}_K \leq \rho^\star$ instead and let the multiplier follow in the manner of a differential penalty (Platt & Barr, 1987). Every 20 updates the noise-free encoder runs for $t_0 = 10$ steps, its state is perturbed by a δ uniform in direction with $\|\delta\| = 0.1\|h_{t_0}\|$, both copies replay the same words for K steps, and

$$\hat{\rho}_K = \text{median}\,\frac{\|\Delta h_{t_0+K}\|}{\|\delta\|}, \qquad \sigma \;\leftarrow\; \text{clip}\big(\sigma \exp(\eta\,(\hat{\rho}_K - \rho^\star)),\, 10^{-4},\, 1\big), \tag{7}$$

with $\eta = 1$ and $\rho^\star = 0.15$, which is Equation (3) read as exponentiated dual ascent on that constraint, clipped so that a transient in $\hat{\rho}_K$ can neither switch the constraint off nor destroy the state. The residual is the right control signal because it is scale-free, needs no actions, rewards or labels (Miller & Hardt, 2019), and responds where the prediction loss stays flat in σ until the noise erases causal directions (Proposition 4, Section 4). The median is a proxy, since the slowest directions of the K-step map survive on every model we trained, so what is held fixed is a typical pace of forgetting rather than a uniform contraction. ACSI therefore adds exactly one quantity a user must choose, the target $\rho^\star$, whose default is conservative rather than optimal, with Corollary 1 giving its feasible range and Section 4 exhibiting a task where that range is empty, and the controller holds $\hat{\rho}_K$ in $[0.073, 0.151]$ over fourteen tasks while $\sigma^\star$ spans more than two orders of magnitude (Section 4).

Output: a discrete read-out. A state close to minimal is close to finite, since $\mathcal{C}$ is finite and Theorem 2 says a recurrent encoder of bounded width represents that partition exactly (Appendix B.8), so a code book fitted on the frozen state turns ACSI into an explicit finite-state abstraction in the sense of Li et al. (2006), one a policy reads as a single symbol instead of a vector it must first partition, which is what lowers the learning burden downstream. Vector quantization (van den Oord et al., 2017) is applied to the frozen encoder states after training, and the policy reads the one-hot indicator of

$$c(h) = \arg\min_{k \leq M} \|h - \mu_k\| \tag{8}$$

in place of the continuous state, the fitting and the choice of M being a protocol detail (Appendix D). This completes ACSI, whose state is learned from action-free data yet distinguishes only the situations that matter, so that a downstream policy spends its experience on those situations alone, and the read-out solves both memory tasks in 24 of 24 runs (Section 4 and Appendix E.2).

4 Experiments

The experiments address four questions in order of importance, namely whether ACSI recovers the causal state from words alone, whether that state makes control easier, whether it carries over to a real stream without ground truth, and whether it works for the reasons Section 2 gives.

We evaluate ACSI, its no-noise ablation and its discrete read-out on a synthetic benchmark with exact ground truth and a real robot stream without it, against strong and recent baselines. We build BUNNIESCAPE (Bunny Escape Adventure), a family of grid tasks whose causal states are enumerated exactly (Table 5 and Figure 5), for the evaluation. NCLT is a real laser stream, read without odometry, with no ground truth. The baselines cover action-conditioned, contrastive, self-predictive, world-model, compression, action-free and latent-action methods, all re-implemented on the same words, GRU ($d = 128$), update budget and checkpoint criterion (Table 9, details in Appendices D and D.2).

Does ACSI recover the causal state? The theory predicts that the noise removes only what prediction does not need, so sufficiency should be untouched, and Table 1, which averages each Bunniescape column over all tasks of Table 7 and the RL score over the tasks on which each row

Table 1: Benchmark against exact ground truth on Bunniescape and NCLT. Every entry averages all Bunniescape tasks or the NCLT sessions, the groups holding the classic baselines, the recent ones and ours. Bold marks the best entry of each column, where SPREAD and codes_{95} compete only among rows within 0.02 AUC of the best, so that a small state counts only if it is still sufficient, and a lower bound ($>$) never wins. † reads odometry actions, since the method is defined only with them.

	Bunniescape (average over all tasks)					Real laser data			
Method	$\lvert S\rvert/\lvert S^\star\rvert \to 1$	AUC ↑	SPREAD ↓	RL score ↑	frames@τ / time@τ	codes_{95} ↓	official AUC ↑	aliasing AUC ↑	recall@1 ↑
Causal-state RNN (Zhang et al., 2019)	1.73	0.826	0.622	0.81	$>$ 0.96M / 0.1min	320†	0.863†	0.800†	0.064†
CPC (van den Oord et al., 2018)	**1.14**	0.854	0.202	0.79	$>$ 1.09M / 0.1min	8	0.829	0.743	0.084
RePo (Zhu et al., 2023)	1.94	0.739	0.660	0.74	$>$ 1.16M / 0.3min	**21**	0.893	0.796	0.070
Self-predictive ϕ_L (Ni et al., 2024)	1.21	0.590	0.593	0.87	$>$ 1.01M / 0.2min	389	0.828	0.672	0.080
ICVF (Ghosh et al., 2023)	0.53	0.514	0.950	0.75	$>$ 1.26M / 0.3min	**21**	0.895	0.764	0.053
DreamerV3 RSSM (Hafner et al., 2025)	1.75	0.685	0.792	0.74	$>$ 0.98M / 0.3min	43	0.904	0.794	0.038
NextLat (Teoh et al., 2025)	1.19	0.850	0.382	0.83	$>$ 0.99M / 0.2min	27	0.856	0.814	0.057
InFOM (Zheng et al., 2026)	2.10	0.867	0.533	0.93	$>$ 0.89M / 0.2min	85	0.892	0.851	**0.098**
LAOM (Nikulin et al., 2025)	1.87	0.775	0.630	0.81	$>$ 1.09M / 0.4min	213	0.792	0.752	0.047
ACSI (no noise)	1.18	0.958	0.159	0.90	$>$ 0.86M / **0.0min**	53	0.897	0.872	0.086
ACSI (prediction + adaptive noise)	1.21	**0.959**	**0.138**	0.92	$>$ 0.54M / **0.0min**	512	**0.910**	**0.891**	0.080
ACSI (discrete, $k{=}2\lvert S^\star\rvert$)	2.50	0.762	0.476	**1.00**	**0.30M** / 0.1min	512	0.869	0.796	0.013
columns won by our block					best in 7 of 10 columns				

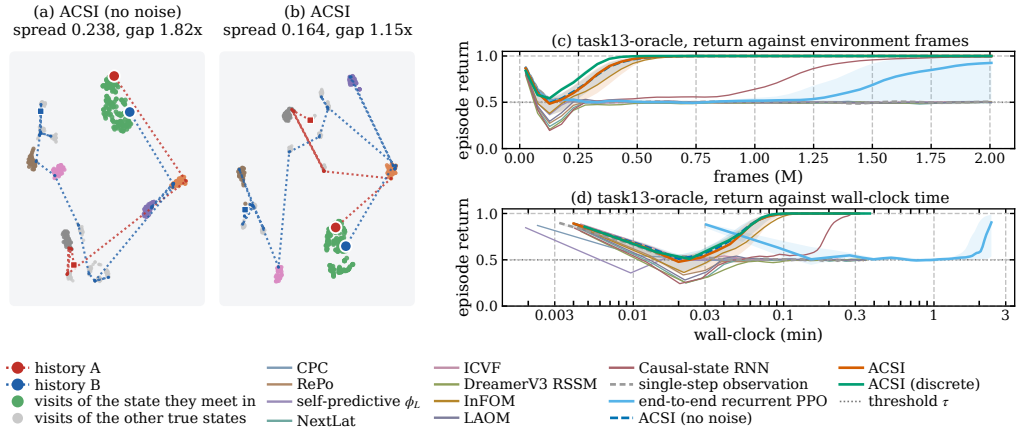


Figure 3: What the recurrent noise merges and how that merging benefits downstream control. (a, b) Held-out representations of `task01-keydoor` coloured by the true minimal state, with two histories of one state dotted, whose endpoint lies 1.15 mean inner distances from that state's cluster under ACSI against 1.82 without noise, closer on 79% of such pairs. (c, d) Median return on `task13-oracle` against frames and wall-clock time (log axis), with bands for our arms, thin lines for the nine baselines of Table 2, and τ marked.

was run (Appendix E.1), confirms this. ACSI has the highest Bunniescape AUC, 0.959 against 0.958 without noise (per task in Table 7), 9 points above the best baseline, InFOM, while methods that compress towards a task or a latent action fall to between 0.514 and 0.775, and the bisimulation baselines DBC (Zhang et al., 2021) and MICo (Castro et al., 2021), which read actions and rewards, stay near chance at 0.524 and 0.543 (Table 7).

The noise acts on minimality instead. It lowers the SPREAD from 0.159 to 0.138 while leaving the size ratio nearly unchanged, from 1.18 to 1.21, the within-state merging that Theorem 1 predicts and panels (a) and (b) of Figure 3 show, and ACSI also leads ARI and AMI on both KeyDoor tasks (Table 8). What it removes is elapsed-time information, since the split of one true state by elapsed time that prediction alone retains falls in silhouette from 0.23 to 0.03 under the noise (Figure 6). Since a representation that also merges different states obtains a small SPREAD trivially, we compare it only within 0.02 AUC of the best, where no baseline qualifies. Baselines with a size ratio closer to one do so at the cost of AUC, such as CPC at 1.14 and 0.854, or undercount the states, as ICVF does (Appendix E.1), and the rest keep distinctions the environment lacks, at ratios of 1.73 to 2.10. This claim rests on Bunniescape, since on NCLT our codes_{95} is the largest (Appendix E.3).

Table 2: Downstream control on the two tasks that require memory. One PPO head reads each frozen state and Table 10 prints the same arms task by task. frames@τ is the median number of frames to reach τ, where a run that never reaches it counts at its full budget, and $>$ marks a median that is only a lower bound. Bold marks the best frozen encoder, and the ground-truth row is in italics.

	task02-keydoor3				task13-oracle			
Policy input	solved	frames@τ	ACSI speed-up	IQM	solved	frames@τ	ACSI speed-up	IQM
Single-step observation	0/24	$>$ 5.01M	$>$ 2.6	2.000	15/24	1.99M	10.0	0.518
End-to-end recurrent PPO	0/24	$>$ 5.01M	$>$ 2.6	2.000	22/24	1.14M	5.7	0.833
Ground-truth state (bound)	*24/24*	*0.50M*	*0.3*	*3.000*	*24/24*	*0.08M*	*0.4*	*1.000*
CPC (van den Oord et al., 2018)	0/8	$>$ 5.01M	$>$ 2.6	2.000	14/24	2.01M	10.1	0.503
RePo (Zhu et al., 2023)	2/8	$>$ 5.01M	$>$ 2.6	2.000	12/24	$>$ 1.99M	$>$ 10.0	0.495
Self-predictive ϕ_L (Ni et al., 2024)	0/8	$>$ 5.01M	$>$ 2.6	2.000	11/24	$>$ 1.99M	$>$ 10.0	0.488
NextLat (Teoh et al., 2025)	0/8	$>$ 5.01M	$>$ 2.6	2.000	11/24	$>$ 2.03M	$>$ 10.2	0.491
ICVF (Ghosh et al., 2023)	0/8	$>$ 5.01M	$>$ 2.6	2.000	10/24	$>$ 1.99M	$>$ 10.0	0.490
DreamerV3 RSSM (Hafner et al., 2025)	5/8	1.91M	1.0	2.750	10/24	$>$ 1.99M	$>$ 10.0	0.491
InFOM (Zheng et al., 2026)	2/8	$>$ 5.01M	$>$ 2.6	2.000	**24/24**	0.20M	1.0	**1.000**
LAOM (Nikulin et al., 2025)	0/8	$>$ 5.01M	$>$ 2.6	2.000	11/24	$>$ 2.03M	$>$ 10.2	0.494
Causal-state RNN (Zhang et al., 2019)	0/8	$>$ 5.01M	$>$ 2.6	2.000	22/24	0.59M	3.0	0.862
ACSI (no noise)	6/24	$>$ 5.01M	$>$ 2.6	2.000	**24/24**	0.18M	0.9	**1.000**
ACSI	13/24	1.91M	1.0	2.583	23/24	0.20M	1.0	**1.000**
ACSI (discrete, $k = 2\|\mathcal{S}^\star\|$)	**24/24**	**0.60M**	**0.3**	**3.000**	**24/24**	**0.08M**	**0.4**	**1.000**

Dropping actions also shrinks the target itself. On the mirror task, where stepping on one cell secretly swaps the effect of two actions, exact enumeration gives 389 interventional classes against 195 causal states, the reduction by half that Proposition 2 predicts, and ACSI follows the smaller partition at no cost in AUC, 0.993 for both encoders, reading the mirror bit at its majority rate of 0.544 where an action-reading encoder stores it at 0.630 (Appendix E.1).

Does the recovered state make control easier? The theory predicts a downstream gain only where control needs memory, and exact computation shows that a memoryless policy already reaches 0.94 to 1.00 of $V^\star$ on seven of ten levels, so we test the two tasks where memory is provably required (Tables 2 and 12 and panels (c) and (d) of Figure 3). ACSI is the only representation that solves both, at 13 of 24 runs in three rooms against 6 of 24 without noise, a probability of improvement of 0.66 with an interval of $[0.53, 0.79]$ (Table 11), and 23 of 24 on the oracle task, while each baseline solves at most one, the DreamerV3 RSSM three rooms and InFOM the oracle task. An end-to-end recurrent policy needs at least 2.6 times the frames in three rooms and 5.7 times on the oracle task, and reading the ACSI state through a code book at $k = 2|S^\star|$ solves 24 of 24 runs on both, matching the ground-truth state (Tables 4 and 13). Where control needs no memory, ACSI is level with its no-noise ablation (Appendix E.2), as the theory expects.

Does it hold on real data? On NCLT, where no ground truth exists, ACSI ranks revisits against perceptually aliased places above every learned baseline, at 0.910 on the official criterion and 0.891 on the aliasing one (all sessions in Table 14). Adding odometry-derived actions to our encoder yields a small gain of 0.013 on the official criterion, as expected of a log driven by a person who need not satisfy Assumption 1, while action-conditioned baselines given the same actions reach an aliasing AUC of only 0.795 and 0.792. Recall at one, a retrieval criterion the causal state is not built for, is won by InFOM and an untrained range histogram (Appendix E.3).

Why does it work? Figure 4 turns the remaining statements of Section 2 into measurements (all five in Table 15). A prediction-only recurrence does not contract geometrically, its $\hat{\rho}_K$ plateauing at 0.146 by $K = 80$ where the noisy one reaches 0.011, the contraction Theorem 1 assumes. Redundancy nothing maintains decays on its own, a consumed bit's probe falling from 0.98 to 0.44 without noise, while redundancy the recurrence rewrites needs the noise, the within-state time probe on `task01-keydoor` reading 0.398, 0.339 after the same further training and 0.130 with noise. The controller is the dual ascent of Proposition 4, holding $\hat{\rho}$ in $[0.073, 0.151]$ over fourteen tasks while $\sigma^\star$ spans $[0.0005, 0.1099]$, and the prediction loss moves 1.6% over the fixed σ that takes the AUC from 0.994 to 0.711, so it cannot be the control signal.

The same measurements fix the design. The state has to be bounded, since without a layer norm the networks escape the noise by growing their states, the root-mean-square state rising from 0.376 to 0.652 on a GRU (Appendix E.4). The noise has to sit in the recurrent channel, since the same σ on

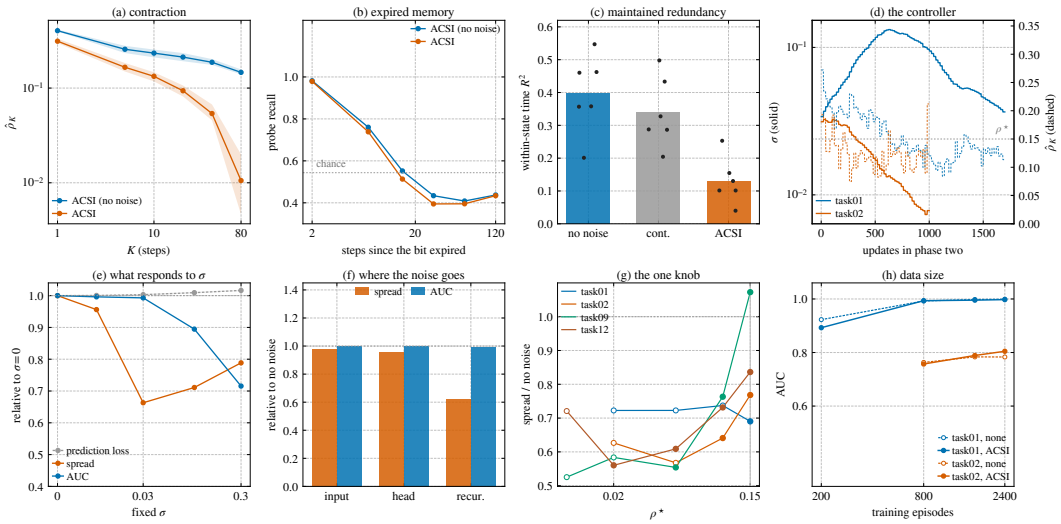


Figure 4: The mechanisms the theory names and the design choices they fix. Panels (a) to (d) read the contraction of Theorem 1, the two redundancies it erases and the dual ascent of Proposition 4, (d) following one seed's σ (solid, left) and $\hat{\rho}_K$ (dashed, right) through phase two, each descending towards $\rho^\star$; panels (e) to (h) fix what responds to the noise, where it sits, the target a user sets and the amount of data from which it becomes effective. Points are seeds, ratios against the same task's prediction-only run, and a hollow marker in (g) has fallen behind it in AUC.

the input embedding or before the head leaves the SPREAD at 0.98 and 0.96 of its no-noise value while the recurrent one takes it to 0.62, and no rate term does that work either. Long memories need the two phases, because a σ large enough to merge erases the key memory when it acts before that memory exists. What transfers is the controller, as the fixed σ that merges one room collapses three rooms in every seed. The target $\rho^\star$ is the only new hyperparameter, its default of 0.15 conservative rather than optimal and holding across four environments, and the noise becomes beneficial once prediction alone stops improving, gaining AUC on three rooms from 1600 episodes upward. How much it helps depends on how slowly the backbone contracts, and where one does not the plug-in supplies the channel (Appendix E.5, with Tables 16 and 17 and Figures 8 and 9).

5 Conclusion and Limitations

Prediction makes a learned state sufficient and forgetting makes it minimal, and a minimal state lets reinforcement learning spend its experience only on the situations that matter. We showed that the causal state of the word process is a well-posed target when actions are not recorded, that the discounted occupancy of future words identifies it, and that a noisy contracting recursion removes what remains while keeping what the future depends on. The strength of that forgetting is set by a feedback rule on a contraction target rather than chosen as a hyperparameter, which is why one target can be reused across tasks that share a prediction horizon while one noise level cannot. Where control provably needs memory this makes control markedly easier, as the only frozen state that solves both such tasks while each baseline solves at most one, at least 2.6 times sooner to threshold in three rooms and 5.7 times sooner on the oracle task than an end-to-end recurrent policy, and at 24 of 24 runs solved on both once the same state is read through a code book.

Limitations. The theory assumes finitely many causal states over a finite alphabet, so continuous observations are quantized before it applies. The learned state is the causal state of the word process under the data policy, so sufficiency for control is supported by our reinforcement learning results rather than by the theory. The target $\rho^\star$ is a single conservative default rather than a universally optimal value. What the controller reads is a median residual, which is a proxy that does not bound the slowest directions of the K-step map. Appendix F states these in detail. Pixel observations, stochastic environments and continuous causal states are where we take the idea next.

AI USE STATEMENT

We used generative AI tools to polish the writing, to help draft sections of the paper, and to help debug code. All research ideas, theory, method design, experiments and analyses were led and carried out by the authors, who reviewed all AI-assisted content and take full responsibility for this work.

ETHICS STATEMENT

This work studies representation learning in simulated grid worlds and on publicly available robot laser logs recorded on a university campus, and it involves no human subjects and no personal data. We see no specific ethical concerns beyond those of reinforcement learning in general.

REPRODUCIBILITY STATEMENT

Appendix D lists all environments, hyperparameters and training details, Appendix D.2 describes every baseline, Appendix B contains all proofs, and Appendix E.3 reads every arm run on the real laser data. The supplementary material contains the code, a unit-test suite, a single entry point that re-runs each experiment by its configuration number, the scripts that compute the exact causal-state partitions and the constructed GRU, and the scripts that generate every figure and table from the logged results.

REFERENCES

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

Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep variational information bottleneck. In *International Conference on Learning Representations (ICLR)*, 2017.

Bowen Baker, Ilge Akkaya, Peter Zhokhov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. Video PreTraining (VPT): Learning to act by watching unlabeled online videos. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, 2022.

Nix Barnett and James P. Crutchfield. Computational mechanics of input–output processes: Structured transformations and the ϵ-transducer. *Journal of Statistical Physics*, 161(2):404–451, 2015. doi: 10.1007/s10955-015-1327-5.

Christopher M. Bishop. Training with noise is equivalent to Tikhonov regularization. *Neural Computation*, 7(1):108–116, 1995. doi: 10.1162/neco.1995.7.1.108.

Jake Bruce et al. Genie: Generative interactive environments. In *International Conference on Machine Learning (ICML)*, pp. 4603–4623, 2024.

Nicholas Carlevaris-Bianco, Arash K. Ushani, and Ryan M. Eustice. University of Michigan north campus long-term vision and lidar dataset. *International Journal of Robotics Research*, 35(9): 1023–1035, 2016.

Mike Casey. The dynamics of discrete-time computation, with application to recurrent neural networks and finite state machine extraction. *Neural Computation*, 8(6):1135–1178, 1996.

Pablo Samuel Castro, Tyler Kastner, Prakash Panangaden, and Mark Rowland. MICo: Improved representations via sampling-based state similarity for Markov decision processes. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021.

Kyunghyun Cho, Bart van Merriënboer, Çağlar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, 2014.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2nd edition, 2006.

Felix Creutzig, Amir Globerson, and Naftali Tishby. Past-future information bottleneck in dynamical systems. *Physical Review E*, 79(4):041925, 2009. doi: 10.1103/PhysRevE.79.041925.

James P. Crutchfield and Karl Young. Inferring statistical complexity. *Physical Review Letters*, 63(2): 105–108, 1989. doi: 10.1103/PhysRevLett.63.105.

Peter Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural Computation*, 5(4):613–624, 1993.

Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite Markov decision processes. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 162–169, 2004.

Carles Gelada, Saurabh Kumar, Jacob Buckman, Ofir Nachum, and Marc G. Bellemare. DeepMDP: Learning continuous latent space models for representation learning. In *International Conference on Machine Learning (ICML)*, pp. 2170–2179, 2019.

Mohammad Gheshlaghi Azar, Rémi Munos, and Hilbert J. Kappen. Minimax PAC bounds on the sample complexity of reinforcement learning with a generative model. *Machine Learning*, 91(3): 325–349, 2013.

Dibya Ghosh, Chethan Anand Bhateja, and Sergey Levine. Reinforcement learning from passive data via latent intentions. In *International Conference on Machine Learning (ICML)*, pp. 11321–11339, 2023.

Robert Givan, Thomas Dean, and Matthew Greig. Equivalence notions and model minimization in Markov decision processes. *Artificial Intelligence*, 147(1–2):163–223, 2003.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640:647–653, 2025.

Michael Janner, Igor Mordatch, and Sergey Levine. γ-models: Generative temporal difference learning for infinite-horizon prediction. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 2020.

Lihong Li, Thomas J. Walsh, and Michael L. Littman. Towards a unified theory of state abstraction for MDPs. In *International Symposium on Artificial Intelligence and Mathematics (ISAIM)*, 2006.

Soon Hoe Lim, N. Benjamin Erichson, Liam Hodgkinson, and Michael W. Mahoney. Noisy recurrent neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021.

Michael L. Littman, Richard S. Sutton, and Satinder Singh. Predictive representations of state. In *Advances in Neural Information Processing Systems (NIPS)*, volume 14, 2001.

Wolfgang Maass and Pekka Orponen. On the effect of analog noise in discrete-time analog computations. *Neural Computation*, 10(5):1071–1095, 1998.

Sarah E. Marzen and James P. Crutchfield. Probabilistic deterministic finite automata and recurrent networks, revisited. *Entropy*, 24(1):90, 2022. doi: 10.3390/e24010090.

John Miller and Moritz Hardt. Stable recurrent models. In *International Conference on Learning Representations (ICLR)*, 2019.

Steven Morad, Ryan Kortvelesy, Matteo Bettini, Stephan Liwicki, and Amanda Prorok. POP-Gym: Benchmarking partially observable reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2023.

Tianwei Ni, Benjamin Eysenbach, and Ruslan Salakhutdinov. Recurrent model-free RL can be a strong baseline for many POMDPs. In *International Conference on Machine Learning (ICML)*, pp. 16691–16723, 2022.

Tianwei Ni, Benjamin Eysenbach, Erfan SeyedSalehi, Michel Ma, Clement Gehring, Aditya Mahajan, and Pierre-Luc Bacon. Bridging state and history representations: Understanding self-predictive RL. In *International Conference on Learning Representations (ICLR)*, 2024.

Alexander Nikulin, Ilya Zisman, Denis Tarasov, Nikita Lyubaykin, Andrei Polubarov, Igor Kiselev, and Vladislav Kurenkov. Latent action learning requires supervision in the presence of distractors. In *International Conference on Machine Learning (ICML)*, 2025.

Antonio Orvieto, Samuel L. Smith, Albert Gu, Anushan Fernando, Çağlar Gülçehre, Razvan Pascanu, and Soham De. Resurrecting recurrent neural networks for long sequences. In *International Conference on Machine Learning (ICML)*, 2023.

Jurgis Pasukonis, Timothy Lillicrap, and Danijar Hafner. Evaluating long-term memory in 3D mazes. In *International Conference on Learning Representations (ICLR)*, 2023.

John C. Platt and Alan H. Barr. Constrained differential optimization. In Dana Z. Anderson (ed.), *Neural Information Processing Systems (NIPS)*, pp. 612–621. American Institute of Physics, 1987.

Danilo Jimenez Rezende and Fabio Viola. Taming VAEs. *arXiv preprint arXiv:1810.00597*, 2018.

Dominik Schmidt and Minqi Jiang. Learning to act without actions. In *International Conference on Learning Representations (ICLR)*, 2024.

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

Max Schwarzer, Ankesh Anand, Rishab Goel, R. Devon Hjelm, Aaron Courville, and Philip Bachman. Data-efficient reinforcement learning with self-predictive representations. In *International Conference on Learning Representations (ICLR)*, 2021.

Younggyo Seo, Kimin Lee, Stephen L. James, and Pieter Abbeel. Reinforcement learning with action-free pre-training from videos. In *International Conference on Machine Learning (ICML)*, 2022.

Cosma Rohilla Shalizi and James P. Crutchfield. Computational mechanics: Pattern and prediction, structure and simplicity. *Journal of Statistical Physics*, 104:817–879, 2001. doi: 10.1023/A:1010388907793.

Cosma Rohilla Shalizi and Kristina Lisa Klinkner. Blind construction of optimal nonlinear recursive predictors for discrete sequences. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 504–511, 2004.

Susanne Still, James P. Crutchfield, and Christopher J. Ellison. Optimal causal inference: Estimating stored information and approximating causal architecture. *Chaos*, 20(3):037111, 2010.

Jayakumar Subramanian, Amit Sinha, Raihan Seraj, and Aditya Mahajan. Approximate information state for approximate planning and reinforcement learning in partially observed systems. *Journal of Machine Learning Research*, 23(12):1–83, 2022.

Richard S. Sutton. Learning to predict by the methods of temporal differences. *Machine Learning*, 3 (1):9–44, 1988.

Yunhao Tang, Zhaohan Daniel Guo, Pierre Harvey Richemond, Bernardo Ávila Pires, Yash Chandak, Rémi Munos, Mark Rowland, Mohammad Gheshlaghi Azar, Charline Le Lan, Clare Lyle, András György, Shantanu Thakoor, Will Dabney, Bilal Piot, Daniele Calandriello, and Michal Valko. Understanding self-predictive learning for reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2023.

Jayden Teoh, Manan Tomar, Kwangjun Ahn, Edward S. Hu, Tim Pearce, Pratyusha Sharma, Akshay Krishnamurthy, Riashat Islam, Alex Lamb, and John Langford. Next-latent prediction transformers learn compact world models. *arXiv preprint arXiv:2511.05963*, 2025.

Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems (NIPS)*, volume 30, pp. 6306–6315, 2017.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Tongzhou Wang, Simon S. Du, Antonio Torralba, Phillip Isola, Amy Zhang, and Yuandong Tian. Denoised MDPs: Learning world models better than the world itself. In *International Conference on Machine Learning (ICML)*, volume 162, pp. 22591–22612, 2022.

Amy Zhang, Zachary C. Lipton, Luis Pineda, Kamyar Azizzadenesheli, Anima Anandkumar, Laurent Itti, Joelle Pineau, and Tommaso Furlanello. Learning causal state representations of partially observable environments. *arXiv preprint arXiv:1906.10437*, 2019.

Amy Zhang, Rowan Thomas McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. In *International Conference on Learning Representations (ICLR)*, 2021.

Chongyi Zheng, Seohong Park, Sergey Levine, and Benjamin Eysenbach. Intention-conditioned flow occupancy models. In *International Conference on Learning Representations (ICLR)*, 2026.

Chuning Zhu, Max Simchowitz, Siri Gadipudi, and Abhishek Gupta. RePo: Resilient model-based reinforcement learning by regularizing posterior predictability. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023.

A Related Work

Causal states and predictive state representations. Computational mechanics defines the causal state as the minimal sufficient statistic of the past for the future together with its update machine, the ϵ-machine (Crutchfield & Young, 1989; Shalizi & Crutchfield, 2001), extends it to input–output processes through the ϵ-transducer (Barnett & Crutchfield, 2015), and reconstructs it classically by splitting histories with statistical tests on discrete sequences (Shalizi & Klinkner, 2004). Representing state by predictions of the future is the same idea in reinforcement learning, from predictive state representations (Littman et al., 2001) and the successor representation (Dayan, 1993) to its infinite-horizon generative form, the γ-model (Janner et al., 2020), and the target of ACSI is the discrete analogue of that occupancy over observation words. ACSI takes the definition and the minimality criterion from this line and supplies what it lacks for deep learning, namely a gradient objective and a mechanism that reaches the minimal statistic rather than merely a sufficient one, which matters because standard recurrent networks fall short of the optimal predictor even on processes generated by probabilistic automata (Marzen & Crutchfield, 2022). Memory benchmarks such as POPGym (Morad et al., 2023) and the 3D Memory Maze (Pasukonis et al., 2023) do not expose the automaton of their environments, so minimality cannot be scored there, whereas our environments make the exact causal-state partition computable.

Bisimulation and task-relevant state abstraction. State abstraction for decision processes (Givan et al., 2003; Li et al., 2006) and bisimulation metrics (Ferns et al., 2004) define equivalence through rewards and action-conditioned transitions, DBC (Zhang et al., 2021) and MICo (Castro et al., 2021) learn such metrics with deep networks, and the closest work to ours, the Causal-state RNN (Zhang et al., 2019), learns causal states of the joint action–observation process and relates them to bisimulation. A parallel line compresses by bounding how much the latent retains about the observation and keeping the controllable and reward-relevant part, since RePo regularizes the predictability of dynamics and reward from the posterior (Zhu et al., 2023) and Denoised MDPs factor the latent into controllable and reward-relevant components and discard the rest (Wang et al., 2022). All of

these target the interventional level, so they need action labels, and rewards decide for them what is relevant, which leaves the reward-driven metrics little to separate states with when rewards are sparse. ACSI works at the observational level instead, and we show that the observational causal state is never finer under a mild condition on the data policy, that it can be strictly smaller, and that it can be learned without action labels.

Self-predictive and world-model latents. Deep reinforcement learning builds predictive latents with recurrent world models (Ha & Schmidhuber, 2018; Hafner et al., 2025), latent-space models (Gelada et al., 2019), contrastive prediction (van den Oord et al., 2018) and self-prediction (Schwarzer et al., 2021), and recurrent model-free agents are strong baselines in partially observable tasks (Ni et al., 2022). Recent theory unifies these objectives as self-predictive abstractions and explains when they avoid collapse (Tang et al., 2023; Ni et al., 2024), and the approximate information state gives planning guarantees for recursively updated sufficient statistics (Subramanian et al., 2022). ACSI shares the recursive statistic and the prediction objective with this line and differs in what it asks of the statistic, because these methods aim at sufficiency, most of them condition on actions, and none of them has a mechanism that removes information which is sufficient but redundant.

Information bottleneck and noise as a rate constraint. The information bottleneck (Tishby et al., 2000) applied to past and future recovers causal states in the zero-distortion limit (Creutzig et al., 2009; Still et al., 2010) and trains neural encoders in its variational form (Alemi et al., 2017), and it is the compression principle that ACSI implements. ACSI implements it through noise in the recurrent state rather than an explicit rate term, which rests on known facts, that input noise acts as a Tikhonov penalty (Bishop, 1995), that hidden-state noise favours stable dynamics (Lim et al., 2021), and that analog noise confines a recurrent network to finite-state computation organized around attractors that mimic the minimal automaton (Casey, 1996; Maass & Orponen, 1998), whereas modern linear recurrences instead initialize their decays close to one to keep long memories (Orvieto et al., 2023). The noise level is set by adapting a Lagrange multiplier, an old idea in neural optimization (Platt & Barr, 1987) that balances reconstruction against compression in VAEs (Rezende & Viola, 2018) and tunes the temperature toward a target entropy in soft actor-critic (Haarnoja et al., 2018), here applied to a new quantity, the contraction of the recurrent map over one prediction horizon measured by replaying words, whose multiplier turns out to be the noise level (Proposition 4). What differs is that the facts above are qualitative and our statement is quantitative, since redundant history is forgotten at a rate set by the contraction while only differences inside a causal state are affected (Theorem 1), and the explicit rate term collapses the representation in our experiments (Section 4).

Action-free and latent-action representation learning. Videos and logs rarely come with actions, so existing approaches infer the missing actions with an inverse dynamics model (Baker et al., 2022), learn latent actions (Schmidt & Jiang, 2024; Bruce et al., 2024; Nikulin et al., 2025), predict the next latent of a world model (Teoh et al., 2025), pre-train action-free video predictors (Seo et al., 2022), or fit value and occupancy models to passive data (Ghosh et al., 2023; Zheng et al., 2026). ACSI takes the same setting and differs in what it recovers, because these objectives return latent actions or the predictive latent of the next step whereas we return the minimal state of the history itself, which needs neither action labels nor a latent-action model and is complementary to them. We run four recent members of these lines as baselines, RePo with its KL bottleneck (Zhu et al., 2023), the self-predictive ϕ_L that Ni et al. (2024) identify as the objective behind that family, the intention-conditioned value of Ghosh et al. (2023), which learns from passive data without actions as we do, and the recurrent state-space model of DreamerV3 (Hafner et al., 2025), which keeps whatever reconstruction needs.

B Proofs

The proofs follow the four steps of Section 2, namely dropping actions (Propositions 1 and 2), sufficiency of prediction and of the occupancy target (Propositions 3 and 6), minimality through the rate bound and multi-step forgetting (Theorem 1), and the calibration of the noise (Proposition 4 and Corollary 1), followed by the GRU construction behind the discrete read-out and a check that our exact evaluation is conservative. Throughout, words $w_t \in \mathcal{W}$ are finite, the environment has finitely many latent states $\mathcal{S}$ and actions $\mathcal{A}$, the next word is emitted deterministically as $w_{t+1} = e(s_t, a_t, s_{t+1})$ (rewards are part of the word), and $T(s' \mid s, a)$ is the transition kernel.

B.1 Sufficiency without actions (Proposition 1)

Proof. Since the history does not record actions, the interventional future after reading w averages over the posterior of the unrecorded action,

$$P(\cdot \mid Hw, \mathrm{do}(\cdot)) = \textstyle\sum_a \rho_H(a \mid w)\, P(\cdot \mid H, \mathrm{do}(a), w, \mathrm{do}(\cdot)),$$
$$\rho_H(a \mid w) \propto \tilde{\pi}(a \mid \iota(H))\, P(w \mid H, \mathrm{do}(a)).$$

Step 1 (right invariance). If $H \sim_{\mathrm{I}} H'$ then $Hw \sim_{\mathrm{I}} H'w$. Interventional equivalence gives the same joint $P(w, \text{rest} \mid H, \mathrm{do}(a), \mathrm{do}(\cdot))$ for H and H', so dividing by the same $P(w \mid H, \mathrm{do}(a))$ gives the same conditional futures, and by Assumption 1 the posteriors $\rho_H = \rho_{H'}$ coincide. *Step 2 (the class is unifilar and carries the next word).* By Step 1 the interventional class updates through a map U, $\iota(Hw) = U(\iota(H), w)$, and the next-word law

$$P_\pi(w_{t+1} = w \mid H) = \textstyle\sum_a \tilde{\pi}(a \mid \iota(H))\, P(w \mid H, \mathrm{do}(a))$$

depends on the history only through $\iota(H)$, since the second factor is the one-step coordinate of the interventional map and the first is Assumption 1. *Step 3 (induction on the length k of the future).* For $k = 0$ there is nothing to show. Let $H \sim_{\mathrm{I}} H'$, so that by Step 2 the two admit the same words with the same probabilities. The observational chain rule, in which no action is conditioned on, gives $P_\pi(w_{t+1:t+k} \mid H) = P_\pi(w_{t+1} \mid H)\, P_\pi(w_{t+2:t+k} \mid Hw_{t+1})$, whose first factor agrees for H and H' by Step 2, while $Hw_{t+1} \sim_{\mathrm{I}} H'w_{t+1}$ by Step 1, so the induction hypothesis at length $k-1$ makes the second factor agree as well. Equal finite-dimensional distributions give $P_\pi(F \mid H) = P_\pi(F \mid H')$. □

Necessity of Assumption 1. Take a single-state environment in which the next word equals the action, $w_{t+1} = a_t \in \{0, 1\}$, and the policy that always repeats the first word, $a_t = w_1$. All histories are interventionally equivalent (one class), but observationally the future is determined by w_1 and there are two classes, so the observational partition is *finer*.

Proposition 5 (Policy memory refines the state)**.** *Let the policy be a finite-state machine that reads words, with memory $m_{t+1} = u(m_t, w_{t+1})$ and actions $a_t \sim \pi(\cdot \mid m_t)$. The causal state is determined by $(\iota(H), \mu_H)$, where μ_H is the posterior over the policy memory given the history. For a deterministic memory, $|\Sigma_{\mathrm{O}}| \le |\Sigma_{\mathrm{I}}| \cdot |\mathcal{M}|$, and if the memory is a function of ι this reduces to Proposition 1.*

Proof. The same induction as above, with the policy index $\iota(H)$ replaced by the pair $(\iota(H), \mu_H)$. The next-word distribution $\sum_{m,a} \mu_H(m)\pi(a \mid m)P(w \mid H, \mathrm{do}(a))$ depends only on this pair. After reading w, the interventional class updates as in Step 1 (with posteriors weighted by μ_H), and the memory posterior updates through u and Bayes' rule, again depending only on the pair. □

The bound is attained only where the memory is visible in the words. Two histories that are interventionally equivalent but carry different memories are separated exactly when the two memories induce different action distributions at some reachable state and the consequences of those actions differ in the emitted word, so the extra states number as many memory states as are told apart on states whose action consequences are visible, and a memory that only steers actions with invisible consequences adds none. Exact computation on the product of the one-room environment and a policy memory confirms the bound, since a hidden per-episode preference (up or down) raises the causal states from 195 to 389, while a memory that is a function of the state ("holds the red key") leaves 195. An encoder trained on such data reads the collector's hidden preference with accuracy 0.983, above the 0.904 achievable from the environment state alone.

Value as a function of the causal state. Under Assumption 1 the value of the behaviour policy is $V^\pi(H) = \mathbb{E}_\pi[\sum_{k \ge 1} \gamma^{k-1} r_{t+k} \mid H]$, which is a functional of $P_\pi(F \mid H)$ because the rewards are part of the words. Observationally equivalent histories therefore have the same value, so V^π is exactly representable as a function of $\sigma(H)$ and an agent evaluating π never needs to distinguish more situations than there are causal states. This is what the value probe of Section 4 measures against, and it does not extend to control, since changing the policy changes $\tilde{\pi}$ in Assumption 1 and with it the partition.

B.2 Strict coarsening of the action-free partition (Proposition 2)

The two partitions differ only when some action sequence separates histories that π leaves indistinguishable, so the causal state is sufficient for control as well wherever they coincide, and Section 4 exhibits a task in which the bit that is dropped is exactly what another policy would need.

In the notation of this appendix, the hypotheses of Proposition 2 read $T(g \cdot s, g \cdot a) = g \cdot T(s, a)$ and $e(g \cdot s, g \cdot a, g \cdot s') = e(s, a, s')$, and its conclusion gives $|\Sigma_{\mathrm{O}}| \leq |\mathcal{S}/G|$, since s and $g \cdot s$ are observationally equivalent.

The condition is sufficient rather than necessary, and it requires the symmetry to be invisible in the words. In a fully observed environment a mirrored layout produces a different observation, so it is not merged.

Proof of Proposition 2. By induction on the length k of the future. For $k = 1$, $P(w \mid s) = \frac{1}{|\mathcal{A}|} \sum_a \mathbf{1}[e(s, a, T(s, a)) = w]$. Substituting $a' = g \cdot a$ (a bijection of $\mathcal{A}$) and using equivariance and invariance term by term gives $P(w \mid s) = P(w \mid g \cdot s)$. For the inductive step, given the first word, the successor sets of s and $g \cdot s$ correspond one-to-one under g because $T(g \cdot s, g \cdot a) = g \cdot T(s, a)$, and the induction hypothesis applies to each pair of successors. With a G-invariant initial distribution the same holds for beliefs, so causal states are unions of orbits. □

Numerical check. We generated 50 random automata (20 base states, 4 actions, 6 observations) equipped with a $\mathbb{Z}_2$ symmetry that flips a hidden bit and swaps two pairs of actions. In all 50, every orbit $\{s, g \cdot s\}$ is merged in the observational partition, the interventional partition is strictly finer, and the ratio of class counts is 2.00 on average and ranges from 1.95 to 2.11, with the deviations coming from base states that are already equivalent. Without such a symmetry strict reduction is rare, arising in 9 of 200 random automata and by at most 3%.

B.3 Sufficiency of exact prediction (classical)

If a recurrent encoder $h_t = f(h_{t-1}, w_t)$ predicts the next word exactly, $q(w \mid h_t) = P_\pi(w_{t+1} = w \mid H_t)$, then $P_\pi(F_t \mid H_t)$ is a function of h_t.

Proof. For every k, $P(w_{t+1:t+k} \mid H_t) = \prod_{j=1}^{k} q\big(w_{t+j} \mid f^{(j-1)}(h_t, w_{t+1:t+j-1})\big)$, where $f^{(j)}$ composes j updates, and each factor depends only on h_t and on the future words themselves. Finite-dimensional distributions determine $P(F_t \mid H_t)$. □

B.4 Occupancy equivalence and the causal-state partition (Propositions 3 and 6)

Let P be the transition matrix of the causal-state machine under the data policy and M the matrix of next-word probabilities, so that the marginal of the k-th future word from state σ is $p_k(\sigma) = e_\sigma^\top P^{k-1} M$ and $m_\gamma(\sigma) = (1-\gamma) e_\sigma^\top (I - \gamma P)^{-1} M$.

Proposition 6 (Occupancy matches causal states)**.** *Let the causal-state machine be finite and* lag-distinguishable*, meaning that for any two distinct causal states $\sigma \neq \sigma'$ there is a $k \leq |\mathcal{C}|$ with $p_k(\sigma) \neq p_k(\sigma')$, equivalently the rows of the observability matrix $O = [M, PM, \ldots, P^{|\mathcal{C}|-1} M]$ are pairwise distinct. Then $m_\gamma(\sigma) - m_\gamma(\sigma')$ is a nonzero rational function of γ, so occupancy equivalence equals the causal-state partition for all but finitely many $\gamma \in (0, 1)$. If the smallest pairwise gap at γ is $s > 0$, the equality holds for all γ' with $|\gamma' - \gamma| < s(1-\gamma)/(4+s)$.*

The two forms of the condition agree because, by Cayley–Hamilton, $v^\top P^{k-1} M = 0$ for $k \leq |\mathcal{C}|$ forces it for every k. Distinct causal states need not satisfy it, since they are distinguished by *some* finite-dimensional distribution, which need not be a single-word marginal. Take $\mathcal{W} = \{0, 1\}$, let A emit 0 to S_0 and 1 to S_1 with probability $\frac{1}{2}$ each, let B emit 0 to S_1 and 1 to S_0, let S_0 emit 0 and S_1 emit 1 to a state C that emits fair coin flips forever. Every lag marginal of A and of B is uniform, so $m_\gamma(A) = m_\gamma(B)$ for every γ, yet $P(00 \mid A) = \frac{1}{2}$ and $P(00 \mid B) = 0$. Occupancy alone would merge them, and Proposition 3 shows that a recurrent encoder does not.

Proof. For $\sigma \neq \sigma'$ lag-distinguishability gives a first $k^\star$ with $p_{k^\star}(\sigma) \neq p_{k^\star}(\sigma')$. The power series $\sum_k \gamma^{k-1}(p_k(\sigma) - p_k(\sigma'))$ therefore has a nonzero coefficient, and it equals the rational function $(I - \gamma P)^{-1}$ applied to $v = e_\sigma - e_{\sigma'}$. A nonzero rational function has finitely many zeros in $(0, 1)$.

For the certificate, write $R = (I-\gamma P)^{-1}$ and $\Delta(\gamma) = (1-\gamma)v^\top RM$. Since $\frac{d}{d\gamma}(I-\gamma P)^{-1} = RPR$, we get $\Delta'(\gamma) = -v^\top RM + (1-\gamma)v^\top RPRM$. Because P and M are row-stochastic and $\|v\|_1 = 2$, $\|v^\top R\|_1 \leq 2/(1-\gamma)$, hence $\|\Delta'(\gamma)\|_1 \leq 4/(1-\gamma)$. If the smallest pairwise gap at γ is s, then for $|\gamma' - \gamma| = \delta$, $\|\Delta(\gamma')\|_1 \geq s - 4\delta/(1-\gamma-\delta) > 0$ whenever $\delta < s(1-\gamma)/(4+s)$. □

Table 3: The exact occupancy partition beside the causal-state partition at four discounts. The two coincide in every column but $\gamma = 0.5$ in three rooms, which is the condition Proposition 6 needs for the target of Section 2.

Task	Reach.	Autom.	Causal	γ=0.5	0.9	0.95	0.99
One room	219	219	195	195	195	195	195
Three rooms	327	327	280	175	280	280	280
Lamp	438	416	389	389	389	389	389
Gap, one room				4.7×10^{-5}	8.1×10^{-3}	1.9×10^{-2}	3.2×10^{-2}
Gap, three rooms				1.5×10^{-14}	9.4×10^{-7}	3.3×10^{-5}	4.0×10^{-3}

The certificate verifies the conclusion of Proposition 6 rather than its hypothesis, since a positive gap s at γ already separates every pair of causal states by occupancy, and hence implies lag-distinguishability, so the condition holds in all our tasks for every $\gamma \geq 0.9$. The partition code was validated on 20 small random automata against brute-force enumeration of 5-step future word distributions (zero disagreements), and on 200 random automata occupancy equivalence equals the causal-state partition for all four discount factors. A spectral-gap condition does not give a useful bound here, because these machines have $|\lambda_2| = |\lambda_3|$, which makes the natural sufficient condition vacuous.

The equivalence a recurrent encoder induces. An encoder that is trained to output m_γ at every step and updates its state deterministically cannot identify two histories unless their occupancies also agree after every continuation, since the same state produces the same prediction from then on. That closure removes the condition.

Proof of Proposition 3. If $\sigma(H) = \sigma(H')$ then unifilarity gives $\sigma(Hu) = \sigma(H'u)$ for every u, and equal causal states have equal futures, hence the same admissible continuations and the same occupancies, so $H \approx_\gamma H'$.

For the converse, let $q_H(w) = P_\pi(w_{t+1} = w \mid H)$ with support S_H. Splitting the occupancy at the first step, $p_k(H) = \sum_w q_H(w) p_{k-1}(Hw)$ for $k \geq 2$ gives

$$m_\gamma(H) = (1-\gamma)\, q_H + \gamma \sum_{w \in S_H} q_H(w)\, m_\gamma(Hw) = A_H(\gamma)\, q_H, \quad A_H(\gamma) = (1-\gamma)\, \iota + \gamma M_H(\gamma)^\top,$$

where ι embeds $\mathbb{R}^{S_H}$ in $\mathbb{R}^{\mathcal{W}}$ and the rows of $M_H(\gamma)$ are the $m_\gamma(Hw)$, $w \in S_H$. The quantities q_H, S_H and $m_\gamma(H)$ depend on H only through $\sigma(H)$, so finitely many such systems occur. Restricted to the rows indexed by S_H, $A_H(\gamma)$ is square with rational entries and equals the identity at $\gamma = 0$, so its determinant is a rational function that is not identically zero and vanishes at finitely many γ. Outside that finite set, every one of these systems determines its q uniquely.

Fix such a γ and let $H \approx_\gamma H'$. Then $S_H = S_{H'}$, the two systems coincide, and $q_H = q_{H'}$. Since $\approx_\gamma$ is closed under appending a word, $Hw \approx_\gamma H'w$ for every $w \in S_H$, and induction on k gives $P_\pi(w_{1:k} \mid H) = q_H(w_1) P_\pi(w_{2:k} \mid Hw_1) = P_\pi(w_{1:k} \mid H')$. Equal finite-dimensional distributions give $H \sim_{\mathrm{O}} H'$. □

In the example above, A and B are separated after the single word 0, which sends A to S_0 and B to S_1, whose occupancies differ.

B.5 Rate bound and minimality (classical)

Rate bound. With $h_t = \mu_t + \varepsilon_t$, $\|\mu_t\|_2 \le B\sqrt{d}$ and $\varepsilon_t \sim \mathcal{N}(0, \sigma^2 I)$ independent of the past, $H_t \to \mu_t \to h_t$ is a Markov chain, so $I(h_t; H_t) \le I(h_t; \mu_t) \le \frac{d}{2}\log(1 + B^2/\sigma^2)$ by the capacity of the parallel Gaussian channel (Cover & Thomas, 2006, Thm. 9.4.1). By Fano's inequality, reading the causal state from h with error at most δ requires $(1-\delta)\log N - 1 \le C$ for N equiprobable states. A single step is far too loose a channel to explain the minimality we measure, since $C \approx 449$ nats for $d = 128$ and $\sigma = 0.03$ against $H(\sigma) < 6$ nats, which is why Section 2 works with noise accumulated over many steps instead. The same cap applies to what the past alone can push through one step, which is the quantity the recurrent channel of Section 3 is meant to restrict. Given w_t we have the Markov chain $H_{t-1} \to h_{t-1} \to h_t$, so $I(h_t; H_{t-1} \mid w_t) \le I(h_{t-1}; H_{t-1} \mid w_t) \le I(h_{t-1}; H_{t-1}) + I(h_{t-1}; w_t \mid H_{t-1})$, and the last term vanishes because given H_{t-1} the state h_{t-1} carries only encoder noise, which is independent of the word the environment emits, so $I(h_t; H_{t-1} \mid w_t) \le C$ as well.

Proof of Lemma 1. Since σ is a function of H, $I(h; H) = I(h; H, \sigma) = I(h; \sigma) + I(h; H \mid \sigma)$. Let $h' \sim p_h(\cdot \mid \sigma(H))$. By construction (h', σ) and (h, σ) have the same joint law and $h' \perp H \mid \sigma$, so $I(h'; H) = I(h'; \sigma) = I(h; \sigma) \le I(h; H)$, with equality iff $I(h; H \mid \sigma) = 0$. Sufficiency of h implies $P(F \mid h) = P(F \mid \sigma(H))$ almost surely, and since distinct causal states have distinct futures, σ is a function of h, hence of h' (same joint law), and $F \perp h' \mid \sigma$ gives $I(h'; F) = I(\sigma; F) = I(H; F)$. □

The redundancy bound of Section 2 follows from the chain rule of Lemma 1 together with $I(h; \sigma) = H(\sigma) - H(\sigma \mid h)$, which gives $I(h; H \mid \sigma) \le C - H(\sigma) + H(\sigma \mid h)$. The residual vanishes exactly when σ is a function of h, and since $h = \mu + \varepsilon$ is Gaussian on $\mathbb{R}^d$, no decoder is error-free, and a decoder with error δ bounds $H(\sigma \mid h) \le h_b(\delta) + \delta \log(N-1)$ by Fano's inequality. The bound is therefore informative in the low-noise regime, in which δ and hence the residual are small.

B.6 Multi-step forgetting (Theorem 1)

Proof. (i) Given the words, $h_t = \mu_t + \varepsilon_t$ with $\mu_t = g(h_s, \varepsilon_{s+1:t-1})$ the recursion run from h_s over $w_{s+1:t}$, and the noise is independent of the words. Couple the two trajectories through the same noise $\varepsilon_{s+1:t-1}$ and through any coupling of the two conditional laws of h_s. Both conditional laws of h_t are then mixtures over the coupling of $\mathcal{N}(\mu_t, \sigma^2 I)$ and $\mathcal{N}(\mu'_t, \sigma^2 I)$, so by joint convexity of the KL divergence $\mathrm{KL} \le \mathbb{E}\,\mathrm{KL}(\mathcal{N}(\mu_t, \sigma^2 I) \| \mathcal{N}(\mu'_t, \sigma^2 I)) = \mathbb{E}\|\mu_t - \mu'_t\|^2/(2\sigma^2)$, which uses no property of f. For the mutual-information bound, fix the shared words $w = w_{s+1:t}$ and read $X \mapsto h_t$ as a channel. Expanding over the channel input,

$$I(X; h_t \mid w_{s+1:t} = w) = \mathbb{E}_x\,\mathrm{KL}\big(P_{h_t|X=x,w} \,\big\|\, P_{h_t|w}\big) \le \max_{x,x'} \mathrm{KL}\big(P_{h_t|X=x,w} \,\big\|\, P_{h_t|X=x',w}\big),$$

by convexity of the KL divergence in its second argument, and each term on the right is a divergence between two such mixtures, hence bounded by the same display. Averaging over w gives the statement.

(ii) Fix the noise sequence. The map carrying the state from step $k-1$ to step k is $h \mapsto f(h, w_k) + \varepsilon_k$, an offset one-step map, so μ_{s+mK} and μ'_{s+mK} are the images of h_s and h'_s under m of the composed block maps of the hypothesis, whose offsets are the noise terms, the offset of the very last step being zero because μ_t excludes ε_t. The hypothesis applies to each block for every realization, so $\|\mu_{s+mK} - \mu'_{s+mK}\| \le \rho_K^m \|h_s - h'_s\|$ pointwise, and squaring and taking expectations gives the stated decay, which substituted into (i) with $\|h_s - h'_s\| \le \mathrm{diam}$ gives the information bound. Between block boundaries the hypothesis says nothing and only (i) applies. The conditioning cannot be dropped, since if X shapes the words read after step s then h_t keeps information about X through those words, and the unconditional $I(X; h_t)$ need not decay. □

The conditioning is not cosmetic for the method either, since a variable fixed before step s can remain legible in h_t through the words it caused after s, and what the noise fades is only the part of that variable which those words do not carry.

B.7 Noise as a multiplier (Proposition 4) and the noise window (Corollary 1)

(a) Linearizing the effect of the noise, $h_t(\varepsilon) \approx h_t(0) + \sum_{s \le t} J_{t,s}\varepsilon_s$. A second-order expansion of $\sum_t \ell_t(h_t)$ under the Gauss–Newton approximation, with $\mathbb{E}\varepsilon_s\varepsilon_{s'}^\top = \sigma^2\delta_{ss'}I$, gives $\mathbb{E}_\varepsilon\mathcal{L} = \mathcal{L}|_{\varepsilon=0} + \frac{\sigma^2}{2}\sum_t\sum_{s\le t}\operatorname{tr}(J_{t,s}^\top G_t J_{t,s}) + o(\sigma^2)$, which is the stated penalty (Bishop, 1995), and Lim et al. (2021) analyse the hidden-state case.

(b) By (a), training with noise σ minimizes, to second order, $\mathcal{L} + \frac{\sigma^2}{2}R$, which is the Lagrangian of $\min_\theta \mathcal{L}$ s.t. $R \le R^\star$ with multiplier $\lambda = \sigma^2/2$. Since $\log\lambda = 2\log\sigma - \log 2$, exponentiated dual ascent $\log\lambda \leftarrow \log\lambda + \eta'(\hat{R} - R^\star)$ is $\log\sigma \leftarrow \log\sigma + \frac{\eta'}{2}(\hat{R} - R^\star)$.

Two substitutions turn this into Equation (3), and neither is an identity. (i) The constraint is moved from R to the measurable $\rho_K^2 \le \rho^{\star 2}$, giving $\log\sigma \leftarrow \log\sigma + \frac{\eta'}{2}(\hat{\rho}_K - \rho^\star)(\hat{\rho}_K + \rho^\star)$, which has the sign of $\hat{\rho}_K - \rho^\star$ and equals Equation (3) up to a positive, bounded step-size factor. The two constraints are not the same function, since R weights each Jacobian direction by the Gauss–Newton matrix G_t of the loss, whereas $\hat{\rho}_K$ perturbs every direction alike, so they disagree exactly on the directions the predictor does not use, which is where the noise acts (see (c)). (ii) In dual ascent the primal variable would also receive ∇_θ of the constraint, whereas in Algorithm 1 it does not, and θ is updated only on the prediction loss under forward noise, while σ alone responds to $\hat{\rho}_K$. Equation (3) is therefore a feedback controller on measured contraction, motivated by this correspondence rather than derived from it. If the response $\rho_K(\sigma)$ of the trained network is continuous and decreasing, the map $\sigma \mapsto \sigma\exp(\eta(\rho_K(\sigma) - \rho^\star))$ has the unique fixed point $\rho_K(\sigma^\star) = \rho^\star$, and standard stochastic-approximation arguments give convergence for small η.

(c) The prediction loss as a signal for σ. Near the optimum the penalty acts mostly along directions that the predictor does not use, where G_t is small, so $\partial\mathcal{L}/\partial\sigma$ stays close to zero until the noise begins to erase causal directions, so a controller driven by $\mathcal{L}$ has no signal to follow whereas ρ_K responds monotonically.

Corollary 1 (Noise window, sufficient conditions)**.** *Let D be the diameter of the representations of one causal state, Δ the distance between causal states, $\rho_K < 1$ the K-step contraction factor of Theorem 1, $L_w < 1$ the contraction inside a basin, and $z_p = \Phi^{-1}(1-p)$ the standard normal quantile. Then $\sigma \ge \rho_K D/\sqrt{2\epsilon}$ suffices to push the bound of Theorem 1 on the retained redundancy below ϵ nats within K steps, and $\sigma \le \Delta\sqrt{1-L_w^2}/(2z_p)$ suffices to tell two causal states apart with error at most p in the linear-Gaussian model of a basin. Some σ satisfies both whenever $\rho_K \cdot D/\Delta \le \sqrt{2\epsilon(1-L_w^2)}/(2z_p)$.*

Proof of Corollary 1. At the lower end, by part (ii) of Theorem 1 over one block, $m = 1$, with the diameter D, the bound $\rho_K^2 D^2/(2\sigma^2)$ on the retained redundancy is at most ϵ iff $\sigma \ge \rho_K D/\sqrt{2\epsilon}$. Since the bound is an upper bound, this is sufficient for ϵ-forgetting and not necessary, since the true retained information may be smaller, so a smaller σ may also do. At the upper end, inside a stable basin with contraction L_w, the stationary spread of the noisy state is $\sigma/\sqrt{1-L_w^2}$ per direction for a linear contraction, and two causal states at distance Δ are told apart with error at most p, where $z_p = \Phi^{-1}(1-p)$, by the optimal two-point Gaussian test iff $\Delta \ge 2z_p\sigma/\sqrt{1-L_w^2}$, which is exact within that linear-Gaussian model and approximate outside it. Some σ satisfies both whenever the lower end does not exceed the upper end, which rearranges to the stated condition. □

The ratio D/Δ in Corollary 1 is the SPREAD of Section 4, and a target read off after K steps is a pace of forgetting per prediction horizon rather than an absolute amount. Because both ends of the window scale with the network's own state, which drifts during training, a fixed σ eventually leaves the window while the scale-free ρ_K does not.

B.8 Representation of the causal state by a GRU (Theorem 2)

Lemma 2 (Unifilarity, classical, Shalizi & Crutchfield 2001)**.** *If two histories have the same causal state, they still do after appending the same word, so $\sigma_t = T(\sigma_{t-1}, w_t)$.*

Theorem 2 (A GRU computes the causal state)**.** *For finitely many causal states $\mathcal{C}$ and a finite alphabet $\mathcal{W}$, there is a single-layer GRU with at most $|\mathcal{C}||\mathcal{W}|+1$ units whose hidden state is one-hot up to $O(e^{-\beta})$ in the weight scale β, so that a linear read-out followed by an argmax returns σ_t exactly on sequences of any length, and the linear read-out alone returns the indicator of σ_t within $O(e^{-\beta})$.*

The construction turns the candidate activation into an AND gate between the previous class and the current word. It bounds the capacity a network needs, 777 units in one room, and the $d = 128$ network we train must instead find a distributed code for the same partition.

Proof of Theorem 2. Write $\mathbf{1}$ for the all-ones vector in $\mathbb{R}^d$ and reserve $\mathbb{1}[\cdot]$ for the indicator. Index the hidden units by the reachable pairs (c, w) plus one start unit, so that $d \le |\mathcal{C}||\mathcal{W}|+1$, with activations within $O(e^{-\beta})$ of $\{\pm 1\}$ and $u = (h+\mathbf{1})/2$ one-hot. Send the update-gate bias to $-\infty$ (so $z \approx 0$) and the reset-gate bias to $+\infty$ (so $r \approx 1$), after which the GRU reduces to $h_t = \tanh(W_{\text{in}} x_t + W_{\text{hn}} h_{t-1} + b)$. Let $A[(c,w),(c',w')] = \mathbb{1}[T(c',w') = c]$ with the start column carrying the initial class, $A[(c,w),\text{start}] = \mathbb{1}[c = c_0]$, let x_t be the one-hot word, and set $W_{\text{hn}} = \frac{\beta}{2}A$, $b_{\text{hn}} = \frac{\beta}{2}A\mathbf{1}$, $W_{\text{in}}[(c,w),w''] = \beta\,\mathbb{1}[w'' = w]$, $b_{\text{in}} = -1.5\beta\,\mathbf{1}$. The start unit has no input weight and an all-zero row in A. Initialize $h_0 = (-1,\dots,-1,+1)^\top$ with the start unit active, so that u_0 is the one-hot start. Since $W_{\text{hn}}h + b_{\text{hn}} = \beta A u$, if h_{t-1} encodes the pair (c_{t-2}, w_{t-1}) then the pre-activation of unit (c,w) equals $\beta(\mathbb{1}[T(c_{t-2},w_{t-1}) = c] + \mathbb{1}[w_t = w] - 1.5)$, which is $+0.5\beta$ only for $(c,w) = (c_{t-1}, w_t)$ and at most -0.5β otherwise; $\tanh$ drives the units to ± 1 with error $O(e^{-\beta})$. At $t = 1$ the same holds with the start column in place of the previous pair. The start unit's own pre-activation is -1.5β at every $t \ge 1$, so it deactivates to -1 after the first word and stays there. The current causal state $\sigma_t = T(c_{t-1}, w_t)$ is a function of the active pair, so the linear read-out $W_{\text{out}}u$ with $W_{\text{out}}[\sigma,(c,w)] = \mathbb{1}[T(c,w) = \sigma]$ returns the indicator of σ_t within $O(e^{-\beta})$, and its argmax returns σ_t exactly. □

The construction script writes these weights into `torch.nn.GRU` with $\beta = 20$ and finite gate biases ∓ 50 in place of $\mp\infty$, sets h_0 as above and reads out by argmax, and Table 4 reports the resulting recovery, which is exact at every step in all our KeyDoor variants. Trained GRUs with 128 units reach an AUC of 0.99 on the same task, so gradient descent approaches the realizable optimum without reproducing the construction.

Table 4: Recovery of the constructed GRU. The construction of Theorem 2 is written into a standard GRU layer with $\beta = 20$ and read back over 200 random episodes of 200 steps.

Task	Reachable beliefs	Causal states	Unifilarity violations	Recovered	Hidden units
One room	197	196	0	100%	777
Mirror	393	196	0	100%	777
Lamp	393	390	0	100%	1527

Every count in this table includes the post-termination absorbing state, which the partition counts of Table 3 and Table 5 exclude. Dropping it leaves 195, 389 and 389 live beliefs against 195, 195 and 389 causal states in one room, the mirror and the lamp, which are the numbers Appendix E.1 quotes, so a mirror belief is finer than a mirror causal state while a lamp belief is not.

B.9 Conservatism of the exact evaluation

Proposition 7 (Conservative ground truth)**.** *Let $\sim_M$ be the minimal-automaton partition of a deterministic environment (initial blocks by the current observation, refined by every action) and let the data policy be uniform. Then $\sim_M \Rightarrow \sim_{\text{I}} \Rightarrow \sim_{\text{O}}$. Hence a pair judged "different" by $\sim_M$ may be observationally equivalent, so false splits are true errors, but some apparent false merges are correct merges, and merging scores computed against $\sim_M$ are conservative.*

Proof. Two states in the same $\sim_M$ block produce the same observation sequence under every action sequence, so they are interventionally equivalent, and the second implication is Proposition 1. □

In our tasks the difference between $\sim_M$ and $\sim_{\text{O}}$ consists only of absorbing states after termination (24, 47 and 27 extra classes in one room, three rooms and lamp), which do not affect any decision.

C Algorithm and Complexity

Algorithm 1 states the procedure of Section 3 in full, phase 1 training the encoder on the occupancy target and phase 2 adding the recurrent noise and its controller, while the discrete read-out is fitted to the frozen encoder afterwards.

Algorithm 1 Action-free Causal State Inference.

Require: word sequences with no actions, target $\rho^\star$, step size η, horizon $K = 1/(1-\gamma)$

1: **Phase 1.** Train encoder and head on the occupancy target (Equation (5)) without noise and keep the best parameters
2: **Phase 2.** restart from those parameters, $\sigma \leftarrow \sigma_0$
3: **repeat**
4: sample episodes, run $h_t \leftarrow f_\theta(h_{t-1}, Ew_t) + \varepsilon_t$, $\varepsilon_t \sim \mathcal{N}(0, \sigma^2 I)$, update θ on the occupancy loss
5: **if** every 20 updates **then**
6: run the noise-free encoder for t_0 steps and perturb h_{t_0} by δ, uniform in direction, $\|\delta\| = 0.1\|h_{t_0}\|$
7: replay the same words for K steps from both states without noise, $\hat{\rho}_K \leftarrow \mathrm{median}_{\mathrm{batch}} \|\Delta h_{t_0+K}\| / \|\delta\|$
8: $\sigma \leftarrow \mathrm{clip}\big(\sigma \exp(\eta(\hat{\rho}_K - \rho^\star)),\, 10^{-4},\, 1\big)$ ▷ Equation (3)
9: **until** early stopping on the held-out prediction loss, evaluated without noise and suspended for the first 300 updates

Complexity. Per update the encoder costs $O(Td^2)$ for a GRU of width d over T steps, and the occupancy head costs $O(Td|\mathcal{W}|)$ over the alphabet $\mathcal{W}$. The controller adds one replay of K steps on the same batch every 20 updates, which multiplies the recurrent cost by a factor $(1 + K/(20T))$, and phase two repeats the update loop, so the method costs the prediction cost times a constant that does not grow with the task. The measurement bears this out. One 20-step replay on the current minibatch once every 20 updates adds a fraction of a percent to the step time, whereas phase two roughly doubles the number of updates, since averaged over 6 seeds ACSI uses 2983 updates against 1617 for ACSI (no noise) in one room, a factor of 1.8, and 3683 against 2600 in three rooms, a factor of 1.4. The noisy forward pass is also unrolled step by step rather than through the fused recurrent kernel, because noise enters between steps, which costs a further constant factor on the wall clock.

D Experimental Setup

This appendix gives what is needed to reproduce Section 4, in the order in which they appear there, namely the environments and data, the baselines, the metrics and protocol, and the hyperparameters.

D.1 Environments and Data

The evaluation runs on two families, a set of exactly enumerable puzzle environments and one real laser dataset, and this subsection describes each of them together with the data it supplies and the ground truth it affords.

Bunniescape. Bunniescape holds two families behind one interface, the KeyDoor rooms of `task01`–`task04` and the rule puzzles of `task05`–`task14`, with one row per task in Table 5 and one tabular per task in Appendix E.1. A word is the pair of the next observation and the reward received on reaching it, indexed by exact equality over at most 256 distinct words, the training data are episodes of a uniform random policy each starting from a reset, 20% of them are held out for early stopping, and evaluation uses 500 separate random episodes. A KeyDoor room is a walled grid whose colour-coded doors each pay $+1$ against the matching key, the keys sit in cabinets that look identical, and the observation is the one-hot position together with a 3×3 egocentric view, so which keys are held and which doors stand open is hidden. `task01` is a single 5×5 room with two keys, of whose 400 configurations 219 are reachable and 195 are causal states, and 31 of its 35 observations are aliased. `task02` chains three rooms, which forces the exit key to be carried through

the whole episode (327 reachable configurations, 280 causal states), `task03` adds a lamp bit that changes the appearance of an indicator and nothing else (438 reachable configurations, 389 causal states, since the bit is visible), and `task04` adds a bit, invisible in the observation, that swaps the effect of the left and right actions (the same 438 configurations and 389 interventional classes, but only 195 causal states, Appendix E.1). A rule puzzle is a grid in the style of *Baba Is You* in which three pushable word plates in a line read NOUN IS PROPERTY, so a push that takes a plate out of line deletes the rule it spelled and a push that brings one into line creates one, and the 5×5 egocentric window hides every plate more than two cells away, which makes the rules in force exactly the part of the state that has to be carried. What each level hides is the *hidden* column of Table 5. Every level is enumerated exactly, since breadth-first search over the reachable configurations tabulates successor, reward and termination, Moore refinement of the partition by observation and terminal flag returns the minimal states, and value iteration at a discount of 0.97 gives the optimal actions, which serve the ground truth alone and never enter training. Table 5 reports, for every one of the fourteen tasks, the family, the hidden variable, the configuration and minimal-state counts, the return a memoryless policy attains and which table or figure of this paper uses the task, and Figure 5 draws five levels along an optimal solution.

NCLT. The real data are the North Campus Long-Term dataset (Carlevaris-Bianco et al., 2016), fifteen months of one campus route driven indoors and outdoors, which is the one arm at a scale where no ground truth is annotated by hand. We use five sessions, 2012-01-08 as the training and database session and 2012-02-05, 2012-06-15, 2013-02-23 and 2013-04-05 as query sessions, and we keep every eighth scan. A word is built by subsampling each scan to 36 beams, taking log ranges and quantizing into 64 words by k-means, and the encoder reads words alone over windows of 200 steps, never odometry or poses. Two ground truths are available without annotation, the word itself, which says when two frames look alike, and the poses shipped with the dataset, which verify 1 284 045 revisit pairs under the distance and heading thresholds of the dataset. The first gives the aliasing AUC, which asks whether two frames carrying the same word stand at the same place, and the second gives the official AUC together with recall@1 and cross-session AR@1, which ask whether the nearest database state of a query frame is a true revisit, inside the training session and from a later session. Settings are chosen on revisits inside the training session alone, the query sessions taking no part in that choice, and every real-data row is 6 seeds. Appendix E.3 reads the capacity and noise sweep, the untrained geometric controls and the action-conditioned controls in full.

D.2 BASELINES

The conditions that make the comparison fair are stated first and the objective each baseline optimizes on our words after them.

All baselines are implemented in our code base and share everything except the objective and, where stated, the input. The data, the word vocabulary, the encoder architecture (a single-layer GRU with $d = 128$ and layer norm unless stated otherwise), the optimizer (Adam, learning rate 10^{-3}, 64 episodes per batch), the maximum of 3000 updates, early stopping on held-out episodes (evaluated every 100 updates, patience 5, best parameters restored), fresh initialization per round, and the evaluation protocols are identical. Following each method, we implemented the loss that shapes the representation, and those losses are given below. For the methods whose own objective does not predict words (CPC, DBC, MICo, ICVF and the self-predictive ϕ_L), an occupancy head is attached to the *stop-gradient* representation, so that it serves early stopping and evaluation alone and cannot shape the representation. These baselines are therefore checkpointed on our unified criterion, and Table 9 repeats four of them checkpointed on their own validation loss and finds our criterion the more lenient of the two. Hyperparameters specific to each baseline take the values recommended in the original papers where they apply, and we did not tune them further.

Causal-state RNN (Zhang et al., 2019). Causal-state RNN optimizes a next-observation likelihood, so its head predicts the next word given the current action, $q(w_{t+1} \mid h_t, a_t)$, which is our target at $\gamma = 0$. It reads actions and, through the word, rewards, the action entering the encoder as an additional one-hot input, $h_t = f(h_{t-1}, [Ew_t; a_{t-1}])$. On our words the alignment is that a_{t-1} is taken before w_t is seen and is available at step t, whereas a_t is taken after it and produces w_{t+1}, so the head reads an action the encoder has not yet read. Everything else is the shared encoder, the shared budget and the shared checkpoint criterion.

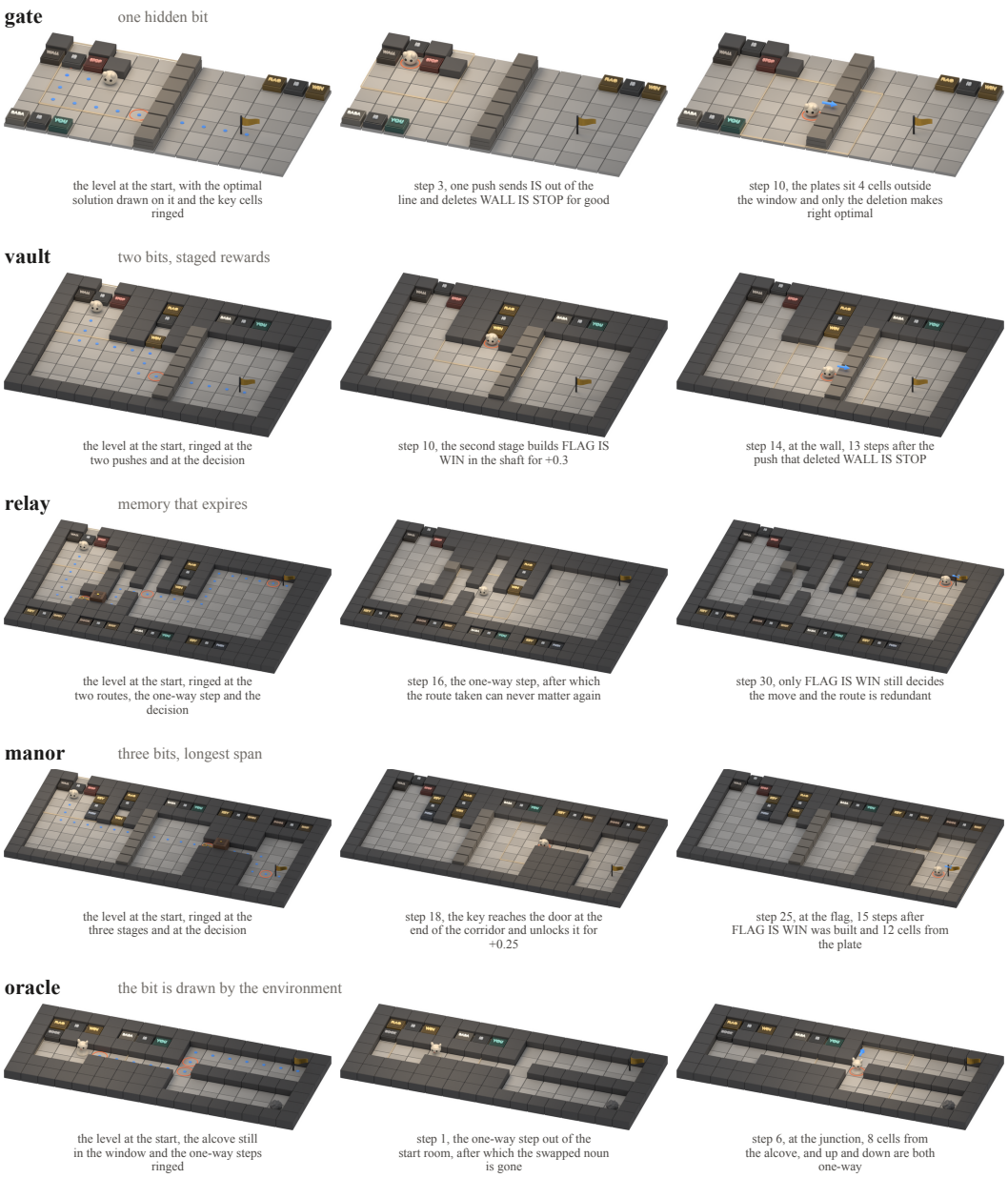


Figure 5: Key states of five levels of the rule puzzle. One level per row and one reachable configuration per panel, the pale rectangle being the 5×5 egocentric window and the arrow of the last panel the optimal action there, which extends Table 5.

CPC (van den Oord et al., 2018). CPC optimizes an InfoNCE bound between the state and the words that follow it, and it reads neither actions nor rewards beyond the reward carried inside the word. On our words a bilinear score $h_t^\top W_k e(w_{t+k})$ with a learned word embedding e is trained for each horizon $k = 1, \ldots, 8$ over 1024 random anchors per step, the remaining anchors serving as negatives and those whose target is the same word being excluded. Its objective does not predict words, so the occupancy head that selects its checkpoint reads a stop-gradient copy of the state.

DBC (Zhang et al., 2021). DBC optimizes a bisimulation metric built from rewards and one-step dynamics, and it reads both, a reward head $\hat{r}(z, a)$ and a Gaussian dynamics head $(\mu, \sigma)(z, a)$ reading the representation and the action while the encoder itself reads words alone. Those heads

Table 5: Every Bunniescape task, what it hides and where it is used in this paper. *Conf. / min.* counts the reachable configurations of a level against the minimal causal states left by Moore refinement and *mem-less* is the memoryless return of Table 12 as a fraction of the optimal one. ‡ `task04` hides its mirror bit from the observation, so its 389 interventional classes leave only 195 action-free causal states (Proposition 2 and Appendix E.1).

Task	Family	Hidden	Conf. / min.	Mem-less	Used in
task01-keydoor	KeyDoor	which of the two keys is carried	219 / 195	–	main, per-task, RL, mechanisms (counters), sensitivity, clustering, t-SNE
task02-keydoor3	KeyDoor	the exit key, carried through the whole episode	327 / 280	–	main, per-task, RL, discrete, sensitivity, clustering
task03-lamp	KeyDoor	a lamp bit that never affects the future	438 / 389	–	setup and proofs only
task04-mirror	KeyDoor	a left–right mirror bit that swaps two actions	438 / 195‡	–	mechanisms (action symmetry)
task05-gate	rule puzzle	wall rule deleted	111 / 111	0.961	main, per-task, RL
task06-swap	rule puzzle	who WIN belongs to	222 / 221	1.000	main, per-task, RL
task07-hazard	rule puzzle	skull rule deleted	87 / 86	0.994	main, per-task, RL
task08-combo	rule puzzle	wall rule, who WIN belongs to	297 / 296	0.978	main, per-task
task09-vault	rule puzzle	wall rule, flag rule, dead switch	528 / 264	0.939	main, per-task, RL, sensitivity
task10-vault_long	rule puzzle	wall rule, flag rule, dead switch	656 / 328	0.686	main, per-task
task11-manor	rule puzzle	wall rule, flag rule, door, 2 dead switches	1844 / 461	0.936	main, per-task
task12-relay	rule puzzle	route taken, which expires, then flag rule	553 / 279	0.997	main, per-task, RL, mechanisms (forgetting, $\rho^\star$), sensitivity
task13-oracle	rule puzzle	who WIN belongs to, drawn by the environment	80 / 80	0.500	main, per-task, RL, discrete
task14-oracle_push	rule puzzle	who WIN belongs to (drawn), wall rule	128 / 126	0.667	memoryless check and this table only

are trained by reward regression and by the log-likelihood of the next stop-gradient representation. On our words the representation is trained so that, for random pairs (i, j), $\|z_i - z_j\|_1$ matches $|\hat{r}_i - \hat{r}_j| + \gamma W_2\big(\mathcal{N}(\mu_i, \sigma_i), \mathcal{N}(\mu_j, \sigma_j)\big)$ with stop-gradient targets, as in the original method. Its occupancy head is detached as for CPC.

MICo (Castro et al., 2021). MICo optimizes a match between a representation distance and a reward-based target, and it reads rewards and, through the sampled transitions, the actions of the data policy. On our words the distance $d(z_x, z_y) = \frac{1}{2}(\|z_x\|^2 + \|z_y\|^2) + \beta\,\angle(z_x, z_y)$ with $\beta = 0.1$ is regressed onto $|r_x - r_y| + \gamma\, d(\operatorname{sg} z'_x, \operatorname{sg} z'_y)$ over random pairs of transitions, where $\angle(z_x, z_y) = \arccos\big(\langle z_x, z_y\rangle / (\|z_x\|\|z_y\|)\big)$ is the angle between the two vectors, as in Castro et al. (2021). Its occupancy head is detached, so it selects the checkpoint without shaping the representation.

RePo (Zhu et al., 2023). RePo optimizes reward and dynamics prediction from a recurrent posterior under a constraint on the KL divergence between posterior and prior, with a multiplier adapted to a KL target of 3.0, and it reads rewards but no actions. Our words already contain the reward, so predicting a separate scalar reward is trivial and its reward head predicts the next word instead, and we normalize the multiplier step, since the unnormalized rule diverges at our loss scale. Table 8 carries the published multiplier setting and the setting that replaces the one-step prediction by our occupancy target. Its state is a sample from the posterior, and the posterior mean is read instead in Table 1.

Self-predictive ϕ_L (Ni et al., 2024). The ϕ_L objective optimizes the prediction of a representation's own next value together with a reward head, with the encoder detached on the target side, and it reads rewards but no actions. Their reward predictability arrives through the downstream critic, and we train representations offline where there is no critic, so a run with the reward coefficient at zero leaves only the self-prediction term and collapses, and we therefore keep the reward head, which is the more favourable of the two options. On our words the representation is the recurrent state at our shared width and budget, and its occupancy head is detached.

ICVF (Ghosh et al., 2023). ICVF optimizes a low-rank bilinear intention-conditioned value $V(\phi(s), \psi(g), T(z))$ fitted on passive data, and it reads rewards and never actions. On our words we take ϕ to be the recurrent state itself rather than a separate encoder, so that the comparison is about the objective, and we apply the exponential moving average to the value head alone. Its occupancy head is detached, and Appendix E.1 confirms that the objective is optimized correctly.

DreamerV3 RSSM (Hafner et al., 2025). The recurrent state-space model of DreamerV3 optimizes the reconstruction, dynamics and representation terms of the original over a latent with a categorical posterior, and it reads actions and rewards, where the others of this group read actions only through our target with actions. On our words the observation it reconstructs is the word itself, and the encoder width, the optimizer and the budget are the shared ones. Its state is a posterior sample, so Table 9 reports the posterior-mean read-out beside it and Table 1 reads that mean.

LAOM (Nikulin et al., 2025). LAOM optimizes an inverse dynamics model that infers a latent action from the present and a future observation through a capacity bottleneck and a forward model that predicts the next observation from the state and that latent action, so the latent carries the choice at a step and the state carries everything else. It reads observations alone, never actions or rewards, and the future offset is sampled uniformly up to 10 steps as in the original. On our discrete words the forward model emits a categorical distribution over the next word under a cross-entropy loss, the observation branch is not detached, because it is the recurrent state we shape, the latent action is 8-dimensional and continuous, and the inverse and forward heads are two-layer ReLU6 networks at our width.

NextLat (Teoh et al., 2025). NextLat optimizes a next-token cross-entropy combined with self-prediction in latent space, where a residual dynamics network rolls the state forward for eight steps and is trained against the stop-gradient state and the stop-gradient next-token distribution at each horizon, and it reads rewards inside the word and no actions. On our words the backbone is our GRU rather than a transformer, which the original describes as backbone-independent, the token embeddings are our input projection for the observation and reward pair, the packed-document mask becomes our within-episode mask, and the coefficients are those of the authors' grid-world configuration. The auxiliary loss on the token after next is zero by default in the original and we do not implement it.

InFOM (Zheng et al., 2026). InFOM optimizes an intention-conditioned occupancy, where one latent intention per trajectory is encoded variationally from the mean of the representations along that trajectory and an occupancy head conditioned on that intention is trained with a KL term towards a standard normal prior. It reads rewards inside the word, and we drop actions, as for ICVF, since our protocol never reads them. On our words flow matching has no useful counterpart on a vocabulary of a few hundred discrete symbols, so the occupancy head is categorical and follows our discounted target under the same TD(λ) schedule as our own. We keep the pre-training stage alone, because there is no critic or policy to adapt offline.

Controls on our own encoder. Four variants share the protocol above and isolate one ingredient each. Our target with actions reads the previous action as an additional one-hot input, exactly as Causal-state RNN does, and is trained with our discounted-occupancy target ($\gamma = 0.95$, TD(λ) with warm-up), which separates the effect of reading actions from the effect of the target. The transformer variant replaces the GRU by a causal transformer with $d = 64$, 2 layers, 4 heads and learned positional embeddings under our target, and for the plug-in of Section 3 the recursion T is a GRU cell of the same width, the consistency noise is $\mathcal{N}(0, 0.3^2 I)$ on the layer-normed state and the plug-in loss has weight 1. The capacity control is our ACSI (no noise) encoder at $d = 512$, which tests width alone.

The recurrent variational bottleneck puts a Gaussian posterior on the recurrent state with a KL term to a standard normal, in one variant with a fixed $\sigma = 0.03$ and in one with a learned, state-dependent σ, over the KL weights swept in Table 17.

D.3 Metrics and Protocol

The quantities reported in Section 4 are defined here, together with the checkpoint criterion that every method shares and the reinforcement learning protocol under which the frozen encoders are read.

Metrics. *AUC.* For aliased observations, pairs of visits from different episodes are labelled "same" or "different" by the true state, and the AUC measures how well the distance between their representations separates the two groups. *Key and value probes.* Linear ridge probes are trained on half of the evaluation episodes and tested on the other half, and the value target is the exact V^π of the random policy from dynamic programming. *MC value.* A linear value fitted to observed returns only (no ground truth), compared with the exact V^π. *NMI.* We run k-means on h with k equal to the number of true states. SPREAD. For each aliased observation, the mean distance between visits with the same state divided by the mean distance between visits with different states, averaged over observations. It measures the history a sufficient state still carries, so it is compared only among representations whose AUC is within 0.02 of the best, since a representation that also merges different states obtains a small SPREAD trivially, which is the case of CPC, whose SPREAD of 0.202 is the lowest among the baselines at an AUC of 0.854. *Time* R^2. A linear probe reads the time step, which the true state itself explains only to $R^2 = 0.42$. *Size ratio.* The number of clusters at which the purity of the quantized state saturates, divided by the number of causal states present in the held-out set, written $|S|/|S^\star|$. *RL score.* The interquartile mean of a frozen encoder's final return as a fraction of the task maximum. *frames@*τ *and time@*τ. The frames and the wall-clock time a run needs in order to reach and hold a return just below that maximum. Frames are a censored median over all runs of an arm, a run that never reached τ entering at the frames it was trained for, so a value whose median run is censored is a lower bound and is written with $>$; the time is a median over the runs that reached τ.

Checkpoint selection and read-out. One checkpoint criterion is applied to every method, an occupancy head attached for early stopping, and the RSSM baselines are read from the posterior mean rather than from a posterior sample, the neutrality of both choices being verified in Table 9. The discrete read-out of Equation (8) adds one step after training and changes nothing inside it: a k-means code book is fitted on the frozen states over the same episodes that trained the encoder, and whatever reads the representation receives the one-hot indicator of the nearest centroid instead of the state itself. Its size is either read from the true state count, at $k = |S^\star|$ or $k = 2|S^\star|$, or set by the 95% purity rule, $k = k_{95}$ being the smallest number of clusters at which the purity of the quantized state against the true minimal states on held-out episodes reaches 95%, which is the same rule that defines the codes$_{95}$ column and the size ratio above. On NCLT, where no true state exists, the code book is fitted post hoc on the state of the continuous row and only codes$_{95}$ is reported.

Reinforcement learning. PPO (Schulman et al., 2017) with 256 parallel environments, 128 steps per rollout, 4 epochs of 4 minibatches, clipping 0.1, entropy coefficient 0.01, GAE $\lambda = 0.95$, discount 0.99, and a linearly annealed learning rate. In one room the policy is an MLP $(64, 64)$ with learning rate 2.5×10^{-4} trained for 10^6 frames, and in three rooms every learned state stays at a return of 2 with the default network, so we use an MLP $(256, 256)$ with learning rate 3×10^{-4} for 5×10^6 frames, identical for all inputs. The encoder is trained once on random-policy data and frozen, and at every step it reads the new word and passes its state to the policy. Every KeyDoor row is 6 encoder seeds with 4 policy seeds each, which gives 24 runs, and the observation and ground-truth inputs need no encoder and use the same 24 policy seeds. The nine baseline rows of Table 2 on `task02-keydoor3` are the exception, at 2 encoder seeds with 4 policy seeds each, 8 runs. Every encoder read by that head is the encoder of Table 1, trained on 800 episodes in one room and on 1600 in three rooms. In one room every state input solves all 24 runs. In three rooms the ACSI state solves 13 of 24 runs against 6 of 24 from the ACSI (no noise) state, with a probability of improvement of 0.66 and an interval of $[0.53, 0.79]$, which is the comparison reported in Section 4 and in Table 11, while the ground-truth state solves 24 of 24.

D.4 Hyperparameters and Compute

The settings shared by every encoder are listed first, followed by the two details of the target and of the noise schedule that the table abbreviates and by the hardware on which everything runs.

Hyperparameters. Table 6 lists all encoder settings, which are shared by all tasks, the GRU having been chosen over a causal transformer at an equal budget.

Table 6: Encoder settings shared by all tasks.

Setting	Value
Encoder	GRU, 1 layer, $d = 128$; layer norm on the output
Prediction target	discounted occupancy of future words, $\gamma = 0.95$ ($K = 20$)
TD(λ)	$\lambda = 0.8$, warm-up from $\lambda = 1$ over 200 updates
Optimizer	Adam, learning rate 10^{-3}, 64 episodes per batch, gradient clipping 0.5
Training	at most 3000 updates per phase; validation every 100 updates (noise off, Monte Carlo target); patience 5; the first 300 updates of phase two are excluded
Noise (phase two)	$\sigma_0 = 0.03$; ramped in over 300 updates when σ is fixed, set by the controller otherwise
Controller	every 20 updates; $t_0 = 10$, perturbation 0.1 (relative), $K = 20$; $\eta = 1$; $\rho^\star = 0.15$; $\sigma \in [10^{-4}, 1]$
Episodes	800 (one room, lamp), 1600 (three rooms) unless stated
Seeds	6 (0–5), paired across variants

Occupancy target at the episode boundary. The recursion of Equation (5) runs backwards from the last valid step of each episode. A terminated episode takes the one-hot of an absorbing END word there, so that the target sums to one at every step, whereas a truncated episode takes the model's own prediction at that step, with the gradient stopped. The TD(λ) mixture with $\mathrm{sg}(\hat{m}_{t+1})$ is applied only at steps strictly before the last one, since at the last one the target is already the tail above.

Contraction measurement and the noise schedule. $\hat{\rho}_K$ is measured on the current minibatch of 64 episodes, using those still running at step $t_0 + K$. For each, the state is run without noise for $t_0 = 10$ steps and perturbed by a δ that is uniform in direction (a standard Gaussian rescaled) with $\|\delta\| = 0.1\|h_{t_0}\|$, after which both states replay the same words for $K = 20$ steps *without* noise, so that the ratio measures the contraction of the transition map and not the diffusion of injected noise, and $\hat{\rho}_K$ is the median of $\|\Delta h_{t_0+K}\|/\|\delta\|$ over those episodes, one perturbation each. The same t_0 is used in every task, so in three rooms it probes the dynamics of the first room, and the controller converges nonetheless, which is the controller row of Table 15. With the controller active, σ starts at $\sigma_0 = 0.03$ and moves only through Equation (3), and the 300 updates of Table 6 are a ramp on σ only when σ is fixed, whereas under the controller they instead suspend early stopping and checkpoint selection at the start of phase 2. Validation is always computed with the noise off and a Monte Carlo target, and phase 2 compares against a baseline reset at its own start, so that the loss increase caused by the noise cannot stop it immediately.

Compute. All runs use a single machine with one NVIDIA RTX 4090 GPU, a 32-core CPU and 128 GB of memory.

E Additional Results

This appendix follows the four questions of Section 4 in the same order, one subsection each, and splits the last question into the theoretical predictions and the design choices they fix. Every entry is a mean over 6 seeds with its standard error, 6 seeds on real data and on the runs added later to complete a task set, and a pair that was never run is absent from its table rather than printed empty.

E.1 RECOVERY OF THE CAUSAL STATE

This subsection extends Section 4, first one task at a time and under the metrics of the representation literature, then through the geometry of what the noise merges, the protocol checks behind every row, and the enumeration that shows the action-free target to be smaller.

Bunniescape results per task. Table 7 traces every average of Table 1 to the all tasks behind it.

Table 7: Bunniescape results per task. One tabular per task traces every average of Table 1 to the levels behind it, over 6 seeds or over 3 for the rows added later to complete the task set.

task01-keydoor

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.767 ± 0.012	0.570 ± 0.016	0.753 ± 0.010	0.361 ± 0.022	768	3.92
CPC	0.817 ± 0.015	0.464 ± 0.015	0.287 ± 0.010	0.597 ± 0.085	407	2.07
DBC	0.526 ± 0.004	0.331 ± 0.010	0.986 ± 0.015	0.724 ± 0.082	768	3.92
MICo	0.523 ± 0.002	0.369 ± 0.005	0.996 ± 0.010	0.242 ± 0.007	512	2.61
RePo (Zhu et al., 2023)	0.678 ± 0.005	0.485 ± 0.008	0.706 ± 0.008	0.397 ± 0.057	427	2.18
Self-predictive ϕ_L (Ni et al., 2024)	0.677 ± 0.066	0.460 ± 0.052	0.408 ± 0.091	0.242 ± 0.076	555	2.83
ICVF (Ghosh et al., 2023)	0.510 ± 0.002	0.334 ± 0.004	1.028 ± 0.013	0.055 ± 0.002	149	0.76
DreamerV3 RSSM (Hafner et al., 2025)	0.768 ± 0.014	0.601 ± 0.013	0.656 ± 0.019	0.396 ± 0.041	469	2.39
NextLat (Teoh et al., 2025)	0.813 ± 0.027	0.609 ± 0.036	0.477 ± 0.021	0.368 ± 0.030	469	2.39
InFOM (Zheng et al., 2026)	0.782 ± 0.008	0.605 ± 0.005	0.661 ± 0.013	0.942 ± 0.009	597	3.05
LAOM (Nikulin et al., 2025)	0.736 ± 0.014	0.480 ± 0.013	0.713 ± 0.039	0.337 ± 0.032	768	3.92
ACSI (no noise)	0.993 ± 0.001	0.803 ± 0.008	0.238 ± 0.012	0.661 ± 0.030	299	1.52
ACSI (prediction + adaptive noise)	0.994 ± 0.001	0.887 ± 0.017	0.164 ± 0.009	0.511 ± 0.015	256	1.31

task02-keydoor3

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.714 ± 0.007	0.600 ± 0.007	0.491 ± 0.011	0.368 ± 0.012	392	1.40
CPC	0.677 ± 0.001	0.543 ± 0.008	0.082 ± 0.003	0.447 ± 0.021	192	0.69
DBC	0.557 ± 0.011	0.377 ± 0.005	0.731 ± 0.049	0.636 ± 0.119	853	3.05
MICo	0.524 ± 0.002	0.430 ± 0.005	1.142 ± 0.011	0.246 ± 0.005	427	1.52
RePo (Zhu et al., 2023)	0.628 ± 0.003	0.575 ± 0.002	0.682 ± 0.005	0.421 ± 0.018	392	1.40
Self-predictive ϕ_L (Ni et al., 2024)	0.595 ± 0.004	0.558 ± 0.005	0.095 ± 0.007	0.352 ± 0.016	96	0.34
ICVF (Ghosh et al., 2023)	0.511 ± 0.001	0.427 ± 0.004	0.935 ± 0.007	0.176 ± 0.007	107	0.38
DreamerV3 RSSM (Hafner et al., 2025)	0.635 ± 0.004	0.576 ± 0.014	0.749 ± 0.013	0.403 ± 0.012	555	1.98
NextLat (Teoh et al., 2025)	0.699 ± 0.012	0.613 ± 0.012	0.306 ± 0.017	0.389 ± 0.013	235	0.84
InFOM (Zheng et al., 2026)	0.735 ± 0.006	0.577 ± 0.007	0.555 ± 0.016	0.901 ± 0.011	597	2.13
LAOM (Nikulin et al., 2025)	0.677 ± 0.005	0.586 ± 0.008	0.593 ± 0.015	0.329 ± 0.015	768	2.74
ACSI (no noise)	0.784 ± 0.015	0.622 ± 0.009	0.167 ± 0.009	0.533 ± 0.031	299	1.07
ACSI (prediction + adaptive noise)	0.789 ± 0.013	0.654 ± 0.010	0.128 ± 0.007	0.470 ± 0.014	213	0.76

task03-lamp

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.820 ± 0.010	0.418 ± 0.002	0.735 ± 0.005	0.299 ± 0.023	768	1.97
CPC	0.908 ± 0.048	0.355 ± 0.014	0.305 ± 0.041	0.573 ± 0.071	768	1.97
DBC	0.520 ± 0.001	0.238 ± 0.000	1.099 ± 0.020	0.840 ± 0.078	1024	2.63
MICo	0.513 ± 0.000	0.257 ± 0.003	1.001 ± 0.009	0.267 ± 0.018	768	1.97
RePo (Zhu et al., 2023)	0.730 ± 0.014	0.340 ± 0.033	0.737 ± 0.024	0.368 ± 0.054	768	1.97
Self-predictive ϕ_L (Ni et al., 2024)	0.704 ± 0.062	0.312 ± 0.011	0.457 ± 0.121	0.229 ± 0.067	768	1.97
ICVF (Ghosh et al., 2023)	0.511 ± 0.000	0.238 ± 0.003	1.123 ± 0.033	0.063 ± 0.002	192	0.49
DreamerV3 RSSM (Hafner et al., 2025)	0.696 ± 0.037	0.356 ± 0.049	0.807 ± 0.023	0.396 ± 0.072	768	1.97
NextLat (Teoh et al., 2025)	0.887 ± 0.007	0.510 ± 0.011	0.490 ± 0.031	0.397 ± 0.013	896	2.30
InFOM (Zheng et al., 2026)	0.813 ± 0.023	0.458 ± 0.025	0.661 ± 0.028	0.946 ± 0.007	1024	2.63
LAOM (Nikulin et al., 2025)	0.776 ± 0.004	0.331 ± 0.021	0.661 ± 0.010	0.305 ± 0.076	768	1.97
ACSI (no noise)	0.961 ± 0.008	0.563 ± 0.014	0.312 ± 0.010	0.681 ± 0.013	TBF	TBF
ACSI (prediction + adaptive noise)	0.975 ± 0.011	0.726 ± 0.039	0.220 ± 0.015	0.519 ± 0.026	597	1.53

task04-mirror

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.724 ± 0.006	0.347 ± 0.006	0.814 ± 0.006	0.345 ± 0.009	1024	2.63
CPC	0.716 ± 0.024	0.258 ± 0.010	0.313 ± 0.076	0.640 ± 0.109	768	1.97
DBC	0.508 ± 0.001	0.195 ± 0.001	1.077 ± 0.027	0.702 ± 0.153	1024	2.63
MICo	0.515 ± 0.000	0.221 ± 0.000	0.991 ± 0.010	0.255 ± 0.025	768	1.97
RePo (Zhu et al., 2023)	0.631 ± 0.016	0.295 ± 0.008	0.835 ± 0.014	0.486 ± 0.068	1024	2.63
Self-predictive ϕ_L (Ni et al., 2024)	0.710 ± 0.106	0.305 ± 0.052	0.595 ± 0.244	0.175 ± 0.102	896	2.30
ICVF (Ghosh et al., 2023)	0.509 ± 0.001	0.198 ± 0.002	1.070 ± 0.053	0.056 ± 0.004	128	0.33
DreamerV3 RSSM (Hafner et al., 2025)	0.750 ± 0.012	0.329 ± 0.002	0.759 ± 0.009	0.247 ± 0.048	768	1.97
NextLat (Teoh et al., 2025)	0.807 ± 0.024	0.355 ± 0.025	0.506 ± 0.056	0.444 ± 0.002	640	1.64
InFOM (Zheng et al., 2026)	0.744 ± 0.012	0.332 ± 0.004	0.678 ± 0.014	0.944 ± 0.012	896	2.30
LAOM (Nikulin et al., 2025)	0.710 ± 0.024	0.266 ± 0.015	0.700 ± 0.006	0.370 ± 0.089	896	2.30
ACSI (no noise)	0.874 ± 0.001	0.430 ± 0.008	0.266 ± 0.016	0.689 ± 0.042	TBF	TBF
ACSI (prediction + adaptive noise)	0.868 ± 0.004	0.484 ± 0.007	0.213 ± 0.017	0.577 ± 0.045	427	1.09

task05-gate ($|\mathcal{S}^\star| = 111$, 111 configurations)

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.999 ± 0.001	0.918 ± 0.007	0.487 ± 0.007	0.144 ± 0.005	112	1.01
CPC	1.000 ± 0.000	0.916 ± 0.006	0.053 ± 0.001	0.190 ± 0.012	96	0.87
DBC	0.558 ± 0.013	0.803 ± 0.025	1.272 ± 0.100	0.902 ± 0.044	384	3.47
MICo	0.714 ± 0.047	0.817 ± 0.018	0.713 ± 0.048	0.649 ± 0.013	384	3.47
RePo (Zhu et al., 2023)	0.990 ± 0.001	0.891 ± 0.009	0.464 ± 0.006	0.233 ± 0.007	122	1.10
Self-predictive ϕ_L (Ni et al., 2024)	0.850 ± 0.095	0.965 ± 0.019	0.316 ± 0.176	0.124 ± 0.009	139	1.25
ICVF (Ghosh et al., 2023)	0.525 ± 0.003	0.909 ± 0.005	1.038 ± 0.010	0.097 ± 0.003	75	0.67
DreamerV3 RSSM (Hafner et al., 2025)	0.879 ± 0.021	0.821 ± 0.008	0.760 ± 0.017	0.198 ± 0.011	213	1.93
NextLat (Teoh et al., 2025)	1.000 ± 0.000	0.958 ± 0.002	0.169 ± 0.008	0.195 ± 0.015	96	0.87
InFOM (Zheng et al., 2026)	0.999 ± 0.001	0.906 ± 0.005	0.341 ± 0.047	0.853 ± 0.034	149	1.35
LAOM (Nikulin et al., 2025)	0.998 ± 0.002	0.903 ± 0.006	0.400 ± 0.063	0.224 ± 0.029	112	1.01
ACSI (no noise)	1.000 ± 0.000	0.935 ± 0.005	0.076 ± 0.007	0.561 ± 0.074	106	0.96
ACSI (prediction + adaptive noise)	1.000 ± 0.000	0.951 ± 0.005	0.082 ± 0.007	0.457 ± 0.072	111	1.00

task06-swap ($|\mathcal{S}^\star| = 221$, 222 configurations)

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.841 ± 0.027	0.926 ± 0.009	0.583 ± 0.001	0.065 ± 0.010	171	1.02
CPC	0.856 ± 0.032	0.932 ± 0.009	0.089 ± 0.006	0.105 ± 0.005	149	0.89
DBC	0.506 ± 0.006	0.756 ± 0.023	1.080 ± 0.079	0.663 ± 0.107	555	3.30
MICo	0.550 ± 0.018	0.886 ± 0.010	1.013 ± 0.042	0.459 ± 0.075	512	3.05
RePo (Zhu et al., 2023)	0.696 ± 0.015	0.918 ± 0.008	0.498 ± 0.026	0.147 ± 0.001	256	1.52
Self-predictive ϕ_L (Ni et al., 2024)	0.504 ± 0.002	0.914 ± 0.013	0.797 ± 0.033	0.062 ± 0.004	96	0.57
ICVF (Ghosh et al., 2023)	0.510 ± 0.004	0.927 ± 0.009	0.992 ± 0.017	0.069 ± 0.004	96	0.57
DreamerV3 RSSM (Hafner et al., 2025)	0.619 ± 0.004	0.914 ± 0.012	0.726 ± 0.034	0.130 ± 0.007	213	1.27
NextLat (Teoh et al., 2025)	0.787 ± 0.028	0.936 ± 0.007	0.362 ± 0.023	0.075 ± 0.015	145	0.86
InFOM (Zheng et al., 2026)	0.849 ± 0.047	0.930 ± 0.008	0.511 ± 0.060	0.643 ± 0.094	299	1.78
LAOM (Nikulin et al., 2025)	0.716 ± 0.042	0.924 ± 0.011	0.620 ± 0.042	0.069 ± 0.004	181	1.08
ACSI (no noise)	0.930 ± 0.031	0.936 ± 0.006	0.088 ± 0.005	0.203 ± 0.021	171	1.02
ACSI (prediction + adaptive noise)	0.941 ± 0.038	0.938 ± 0.006	0.111 ± 0.007	0.228 ± 0.037	181	1.08

task07-hazard ($|\mathcal{S}^\star| = 86$, 87 configurations)

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.880 ± 0.011	0.903 ± 0.013	0.581 ± 0.009	0.220 ± 0.005	113	1.33
CPC	0.906 ± 0.042	0.870 ± 0.004	0.088 ± 0.008	0.268 ± 0.007	78	0.92
DBC	0.530 ± 0.005	0.765 ± 0.025	1.145 ± 0.036	0.817 ± 0.075	213	2.51
MICo	0.605 ± 0.011	0.809 ± 0.011	0.818 ± 0.023	0.685 ± 0.039	213	2.51
RePo (Zhu et al., 2023)	0.803 ± 0.004	0.886 ± 0.013	0.547 ± 0.010	0.394 ± 0.006	117	1.38
Self-predictive ϕ_L (Ni et al., 2024)	0.534 ± 0.012	0.808 ± 0.006	0.852 ± 0.089	0.084 ± 0.009	121	1.42
ICVF (Ghosh et al., 2023)	0.529 ± 0.003	0.809 ± 0.005	0.953 ± 0.027	0.112 ± 0.001	53	0.63
DreamerV3 RSSM (Hafner et al., 2025)	0.677 ± 0.022	0.833 ± 0.014	0.815 ± 0.015	0.289 ± 0.035	192	2.26
NextLat (Teoh et al., 2025)	0.946 ± 0.011	0.963 ± 0.007	0.205 ± 0.019	0.240 ± 0.003	78	0.92
InFOM (Zheng et al., 2026)	0.880 ± 0.020	0.874 ± 0.012	0.618 ± 0.034	0.979 ± 0.001	192	2.26
LAOM (Nikulin et al., 2025)	0.887 ± 0.007	0.863 ± 0.008	0.545 ± 0.034	0.272 ± 0.032	99	1.16
ACSI (no noise)	0.970 ± 0.020	0.947 ± 0.008	0.105 ± 0.005	0.670 ± 0.039	85	1.00
ACSI (prediction + adaptive noise)	0.959 ± 0.019	0.986 ± 0.002	0.115 ± 0.011	0.561 ± 0.035	85	1.00

task08-combo ($|\mathcal{S}^\star| = 296$, 297 configurations)

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.899 ± 0.011	0.820 ± 0.010	0.588 ± 0.012	0.235 ± 0.005	384	1.48
CPC	0.973 ± 0.001	0.829 ± 0.012	0.115 ± 0.004	0.332 ± 0.028	235	0.91
DBC	0.541 ± 0.003	0.710 ± 0.028	1.116 ± 0.006	0.880 ± 0.033	768	2.97
MICo	0.551 ± 0.011	0.762 ± 0.006	0.927 ± 0.027	0.577 ± 0.164	683	2.64
RePo (Zhu et al., 2023)	0.871 ± 0.009	0.810 ± 0.011	0.519 ± 0.007	0.374 ± 0.033	384	1.48
Self-predictive ϕ_L (Ni et al., 2024)	0.647 ± 0.002	0.843 ± 0.005	0.066 ± 0.002	0.151 ± 0.001	107	0.41
ICVF (Ghosh et al., 2023)	0.536 ± 0.005	0.810 ± 0.010	0.829 ± 0.046	0.134 ± 0.004	107	0.41
DreamerV3 RSSM (Hafner et al., 2025)	0.743 ± 0.020	0.802 ± 0.013	0.691 ± 0.022	0.288 ± 0.013	427	1.65
NextLat (Teoh et al., 2025)	0.958 ± 0.004	0.835 ± 0.010	0.273 ± 0.012	0.285 ± 0.011	213	0.82
InFOM (Zheng et al., 2026)	0.950 ± 0.012	0.817 ± 0.009	0.506 ± 0.038	0.882 ± 0.020	512	1.98
LAOM (Nikulin et al., 2025)	0.855 ± 0.013	0.806 ± 0.009	0.638 ± 0.008	0.226 ± 0.002	427	1.65
ACSI (no noise)	0.975 ± 0.005	0.846 ± 0.008	0.167 ± 0.004	0.684 ± 0.037	384	1.48
ACSI (prediction + adaptive noise)	0.983 ± 0.006	0.861 ± 0.007	0.140 ± 0.006	0.571 ± 0.040	341	1.32

task09-vault ($|\mathcal{S}^\star| = 264$, 528 configurations)

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.865 ± 0.013	0.783 ± 0.010	0.570 ± 0.008	0.131 ± 0.010	338	1.34
CPC	0.961 ± 0.011	0.778 ± 0.009	0.091 ± 0.005	0.200 ± 0.004	192	0.76
DBC	0.524 ± 0.006	0.711 ± 0.029	1.256 ± 0.012	0.827 ± 0.007	683	2.71
MICo	0.538 ± 0.004	0.731 ± 0.010	1.023 ± 0.034	0.438 ± 0.044	768	3.05
RePo (Zhu et al., 2023)	0.806 ± 0.010	0.781 ± 0.011	0.643 ± 0.008	0.273 ± 0.039	341	1.35
Self-predictive ϕ_L (Ni et al., 2024)	0.512 ± 0.004	0.764 ± 0.010	0.936 ± 0.047	0.054 ± 0.011	255	1.01
ICVF (Ghosh et al., 2023)	0.502 ± 0.004	0.777 ± 0.010	0.995 ± 0.007	0.055 ± 0.004	96	0.38
DreamerV3 RSSM (Hafner et al., 2025)	0.672 ± 0.007	0.736 ± 0.010	0.844 ± 0.008	0.182 ± 0.011	274	1.09
NextLat (Teoh et al., 2025)	0.936 ± 0.008	0.805 ± 0.011	0.330 ± 0.024	0.184 ± 0.023	256	1.02
InFOM (Zheng et al., 2026)	0.939 ± 0.019	0.775 ± 0.009	0.528 ± 0.059	0.793 ± 0.053	381	1.51
LAOM (Nikulin et al., 2025)	0.819 ± 0.014	0.778 ± 0.010	0.564 ± 0.010	0.110 ± 0.006	295	1.17
ACSI (no noise)	0.993 ± 0.004	0.787 ± 0.010	0.122 ± 0.005	0.515 ± 0.046	222	0.88
ACSI (prediction + adaptive noise)	0.991 ± 0.005	0.820 ± 0.012	0.130 ± 0.009	0.391 ± 0.067	231	0.92

task10-vault_long ($|\mathcal{S}^\star| = 328,656$ configurations)

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.954 ± 0.016	0.655 ± 0.017	0.397 ± 0.005	0.330 ± 0.010	427	1.46
CPC	0.921 ± 0.017	0.601 ± 0.020	0.083 ± 0.006	0.355 ± 0.006	256	0.88
DBC	0.508 ± 0.007	0.334 ± 0.004	1.073 ± 0.004	0.320 ± 0.028	853	2.93
MICo	0.511 ± 0.010	0.376 ± 0.005	1.117 ± 0.039	0.110 ± 0.039	853	2.93
RePo (Zhu et al., 2023)	0.659 ± 0.008	0.626 ± 0.013	0.792 ± 0.013	0.348 ± 0.007	1024	3.51
Self-predictive ϕ_L (Ni et al., 2024)	0.504 ± 0.002	0.388 ± 0.008	0.973 ± 0.010	0.052 ± 0.002	235	0.81
ICVF (Ghosh et al., 2023)	0.501 ± 0.003	0.400 ± 0.008	1.010 ± 0.015	0.067 ± 0.003	96	0.33
DreamerV3 RSSM (Hafner et al., 2025)	0.586 ± 0.013	0.507 ± 0.022	0.912 ± 0.011	0.292 ± 0.015	597	2.05
NextLat (Teoh et al., 2025)	0.933 ± 0.023	0.789 ± 0.007	0.219 ± 0.003	0.351 ± 0.004	291	1.00
InFOM (Zheng et al., 2026)	0.936 ± 0.018	0.670 ± 0.024	0.307 ± 0.046	0.540 ± 0.067	384	1.32
LAOM (Nikulin et al., 2025)	0.833 ± 0.022	0.524 ± 0.009	0.589 ± 0.005	0.334 ± 0.009	939	3.22
ACSI (no noise)	0.962 ± 0.010	0.738 ± 0.015	0.069 ± 0.002	0.366 ± 0.003	328	1.13
ACSI (prediction + adaptive noise)	0.960 ± 0.006	0.873 ± 0.015	0.067 ± 0.003	0.380 ± 0.010	291	1.00

task11-manor ($|\mathcal{S}^\star| = 461,1844$ configurations)

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.899 ± 0.008	0.682 ± 0.009	0.479 ± 0.009	0.233 ± 0.015	367	1.11
CPC	0.949 ± 0.014	0.661 ± 0.008	0.069 ± 0.003	0.269 ± 0.012	256	0.77
DBC	0.522 ± 0.002	0.577 ± 0.011	1.189 ± 0.034	0.653 ± 0.167	768	2.32
MICo	0.516 ± 0.005	0.579 ± 0.014	1.030 ± 0.015	0.327 ± 0.028	853	2.57
RePo (Zhu et al., 2023)	0.802 ± 0.019	0.683 ± 0.015	0.650 ± 0.026	0.306 ± 0.024	683	2.06
Self-predictive ϕ_L (Ni et al., 2024)	0.507 ± 0.002	0.636 ± 0.006	0.988 ± 0.013	0.069 ± 0.003	427	1.29
ICVF (Ghosh et al., 2023)	0.506 ± 0.002	0.639 ± 0.006	1.012 ± 0.011	0.076 ± 0.003	107	0.32
DreamerV3 RSSM (Hafner et al., 2025)	0.658 ± 0.014	0.609 ± 0.007	0.838 ± 0.009	0.246 ± 0.007	512	1.54
NextLat (Teoh et al., 2025)	0.926 ± 0.015	0.719 ± 0.018	0.278 ± 0.020	0.238 ± 0.010	256	0.77
InFOM (Zheng et al., 2026)	0.871 ± 0.047	0.661 ± 0.012	0.580 ± 0.037	0.877 ± 0.027	683	2.06
LAOM (Nikulin et al., 2025)	0.825 ± 0.043	0.654 ± 0.007	0.578 ± 0.049	0.209 ± 0.029	427	1.29
ACSI (no noise)	0.971 ± 0.013	0.697 ± 0.009	0.112 ± 0.006	0.352 ± 0.026	306	0.92
ACSI (prediction + adaptive noise)	0.964 ± 0.015	0.766 ± 0.017	0.095 ± 0.008	0.356 ± 0.035	306	0.92

task12-relay ($|\mathcal{S}^\star| = 279,553$ configurations)

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.890 ± 0.015	0.715 ± 0.007	0.610 ± 0.011	0.343 ± 0.012	597	2.14
CPC	1.000 ± 0.000	0.689 ± 0.008	0.087 ± 0.003	0.411 ± 0.018	314	1.13
DBC	0.526 ± 0.004	0.472 ± 0.005	1.072 ± 0.012	0.667 ± 0.154	939	3.36
MICo	0.500 ± 0.000	0.517 ± 0.009	1.035 ± 0.023	0.276 ± 0.016	939	3.36
RePo (Zhu et al., 2023)	0.842 ± 0.018	0.714 ± 0.009	0.573 ± 0.017	0.397 ± 0.006	853	3.06
Self-predictive ϕ_L (Ni et al., 2024)	0.509 ± 0.001	0.536 ± 0.015	0.645 ± 0.005	0.173 ± 0.005	192	0.69
ICVF (Ghosh et al., 2023)	0.507 ± 0.001	0.569 ± 0.005	0.985 ± 0.008	0.193 ± 0.009	192	0.69
DreamerV3 RSSM (Hafner et al., 2025)	0.761 ± 0.019	0.625 ± 0.007	0.752 ± 0.009	0.363 ± 0.015	512	1.84
NextLat (Teoh et al., 2025)	0.906 ± 0.019	0.824 ± 0.011	0.362 ± 0.008	0.388 ± 0.009	349	1.25
InFOM (Zheng et al., 2026)	0.999 ± 0.000	0.762 ± 0.009	0.271 ± 0.019	0.696 ± 0.025	427	1.53
LAOM (Nikulin et al., 2025)	0.821 ± 0.021	0.679 ± 0.010	0.584 ± 0.014	0.347 ± 0.012	683	2.45
ACSI (no noise)	1.000 ± 0.000	0.832 ± 0.011	0.123 ± 0.004	0.518 ± 0.015	384	1.38
ACSI (prediction + adaptive noise)	1.000 ± 0.000	0.953 ± 0.005	0.103 ± 0.004	0.495 ± 0.018	384	1.38

task13-oracle ($|\mathcal{S}^\star| = 80,80$ configurations)

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.623 ± 0.040	0.787 ± 0.045	0.938 ± 0.036	0.463 ± 0.007	155	1.93
CPC	0.503 ± 0.003	0.673 ± 0.009	1.001 ± 0.014	0.478 ± 0.005	69	0.87
DBC	0.506 ± 0.001	0.621 ± 0.034	1.015 ± 0.004	0.651 ± 0.023	192	2.40
MICo	0.531 ± 0.004	0.626 ± 0.009	1.088 ± 0.019	0.600 ± 0.002	213	2.67
RePo (Zhu et al., 2023)	0.507 ± 0.002	0.702 ± 0.009	1.006 ± 0.003	0.535 ± 0.011	139	1.73
Self-predictive ϕ_L (Ni et al., 2024)	0.497 ± 0.002	0.658 ± 0.004	1.075 ± 0.033	0.446 ± 0.002	101	1.27
ICVF (Ghosh et al., 2023)	0.501 ± 0.001	0.657 ± 0.010	0.953 ± 0.003	0.460 ± 0.007	53	0.67
DreamerV3 RSSM (Hafner et al., 2025)	0.527 ± 0.005	0.710 ± 0.028	0.961 ± 0.007	0.543 ± 0.007	80	1.00
NextLat (Teoh et al., 2025)	0.531 ± 0.007	0.665 ± 0.011	1.063 ± 0.012	0.473 ± 0.010	59	0.73
InFOM (Zheng et al., 2026)	0.729 ± 0.049	0.889 ± 0.024	0.772 ± 0.038	0.931 ± 0.002	235	2.93
LAOM (Nikulin et al., 2025)	0.502 ± 0.001	0.664 ± 0.009	1.025 ± 0.005	0.463 ± 0.007	75	0.93
ACSI (no noise)	1.000 ± 0.000	0.950 ± 0.011	0.183 ± 0.006	0.590 ± 0.018	128	1.60
ACSI (prediction + adaptive noise)	1.000 ± 0.000	0.968 ± 0.008	0.214 ± 0.018	0.587 ± 0.016	128	1.60

task14-oracle_push ($|\mathcal{S}^\star| = 126,128$ configurations)

Method	AUC ↑	Probe ↑	SPREAD ↓	Time R^2 ↓	k_{95}	$\lvert S\rvert/\lvert S^\star\rvert$
Causal-state RNN (Zhang et al., 2019)	0.693 ± 0.009	0.827 ± 0.004	0.682 ± 0.021	0.318 ± 0.021	192	1.52
CPC	0.768 ± 0.001	0.855 ± 0.003	0.165 ± 0.027	0.421 ± 0.046	159	1.26
DBC	0.509 ± 0.001	0.625 ± 0.014	0.958 ± 0.004	0.663 ± 0.096	256	2.03
MICo	0.504 ± 0.004	0.723 ± 0.006	0.940 ± 0.010	0.409 ± 0.019	256	2.03
RePo (Zhu et al., 2023)	0.704 ± 0.014	0.818 ± 0.014	0.585 ± 0.025	0.494 ± 0.086	224	1.78
Self-predictive ϕ_L (Ni et al., 2024)	0.517 ± 0.001	0.729 ± 0.009	0.104 ± 0.012	0.252 ± 0.029	96	0.76
ICVF (Ghosh et al., 2023)	0.539 ± 0.001	0.758 ± 0.004	0.379 ± 0.102	0.281 ± 0.025	96	0.76
DreamerV3 RSSM (Hafner et al., 2025)	0.616 ± 0.016	0.763 ± 0.006	0.817 ± 0.001	0.400 ± 0.005	192	1.52
NextLat (Teoh et al., 2025)	0.776 ± 0.002	0.871 ± 0.007	0.315 ± 0.028	0.368 ± 0.000	159	1.26
InFOM (Zheng et al., 2026)	0.910 ± 0.004	0.900 ± 0.010	0.468 ± 0.041	0.877 ± 0.013	320	2.54
LAOM (Nikulin et al., 2025)	0.691 ± 0.024	0.802 ± 0.019	0.611 ± 0.002	0.338 ± 0.012	159	1.26
ACSI (no noise)	0.995 ± 0.001	0.922 ± 0.006	0.194 ± 0.005	0.645 ± 0.036	TBF	TBF
ACSI (prediction + adaptive noise)	0.998 ± 0.000	0.970 ± 0.006	0.144 ± 0.009	0.599 ± 0.016	256	2.03

Standard representation metrics for every method. Table 8 holds the quantities the representation literature reports beside the AUC and the SPREAD, on the same checkpoints and under the shared protocol of Appendix D.2, with five neighbours for k-NN and the posterior mean for the RSSM rows. The k-nearest-neighbour accuracy is saturated in one room, where ACSI and ACSI (no noise) score 0.998 and 1.000, so it separates nothing there. The participation ratio does not fall under ACSI, rising from 6.30 to 6.61 in one room and falling only from 3.41 to 3.28 in three rooms, so the minimality this paper claims is the merging of histories of one causal state and not a reduction of linear dimension. And a low SPREAD on its own is not evidence, since CPC attains 0.082 in three rooms against our 0.128 while recovering the state far worse.

Table 8: Standard representation metrics for every method. They are computed on the same checkpoints as Table 1 and extend the sufficiency and the minimality paragraphs of Section 4.

Method	AUC	Probe	F1	k-NN	ARI	AMI	Ppl.	PR	SPREAD
task01-keydoor (one room), 800 episodes									
ACSI	0.994 ± 0.001	0.887 ± 0.017	0.676 ± 0.020	0.998 ± 0.001	0.841 ± 0.011	0.955 ± 0.002	3.881 ± 0.002	6.606 ± 0.319	0.164 ± 0.009
ACSI (no noise)	0.993 ± 0.001	0.803 ± 0.008	0.582 ± 0.007	1.000 ± 0.000	0.822 ± 0.013	0.954 ± 0.003	3.897 ± 0.005	6.302 ± 0.062	0.238 ± 0.012
Our target + actions	0.993 ± 0.001	0.795 ± 0.015	0.571 ± 0.018	0.999 ± 0.001	0.823 ± 0.011	0.952 ± 0.002	3.903 ± 0.005	6.197 ± 0.172	0.239 ± 0.010
Causal-state RNN (Zhang et al., 2019)	0.767 ± 0.012	0.570 ± 0.016	0.326 ± 0.018	0.892 ± 0.031	0.392 ± 0.015	0.787 ± 0.006	3.914 ± 0.003	16.550 ± 0.240	0.753 ± 0.010
CPC	0.817 ± 0.015	0.464 ± 0.015	0.239 ± 0.014	0.783 ± 0.017	0.485 ± 0.027	0.829 ± 0.010	3.859 ± 0.003	6.265 ± 0.170	0.287 ± 0.010
DBC	0.526 ± 0.004	0.331 ± 0.010	0.125 ± 0.004	0.087 ± 0.052	0.148 ± 0.005	0.516 ± 0.013	6.176 ± 0.097	14.232 ± 3.355	0.986 ± 0.015
MICo	0.523 ± 0.002	0.369 ± 0.005	0.136 ± 0.003	0.379 ± 0.003	0.186 ± 0.003	0.612 ± 0.007	4.429 ± 0.018	4.585 ± 0.668	0.996 ± 0.010
GRU d=512	0.986 ± 0.002	0.900 ± 0.008	0.690 ± 0.008	0.997 ± 0.001	0.770 ± 0.008	0.929 ± 0.002	3.928 ± 0.003	6.764 ± 0.056	0.320 ± 0.005
Transformer	0.835 ± 0.009	0.308 ± 0.007	0.161 ± 0.006	0.748 ± 0.020	0.417 ± 0.020	0.711 ± 0.015	6.283 ± 0.037	5.196 ± 0.215	0.482 ± 0.021
RePo (Zhu et al., 2023)	0.678 ± 0.005	0.485 ± 0.008	0.271 ± 0.006	0.609 ± 0.042	0.344 ± 0.004	0.764 ± 0.003	3.914 ± 0.002	6.367 ± 0.153	0.706 ± 0.008
with our occupancy target	0.953 ± 0.012	0.663 ± 0.023	0.452 ± 0.021	0.966 ± 0.013	0.724 ± 0.028	0.909 ± 0.009	5.461 ± 0.048	4.886 ± 0.143	0.358 ± 0.014
Self-predictive ϕ_L (Ni et al., 2024)	0.677 ± 0.066	0.460 ± 0.052	0.239 ± 0.043	0.537 ± 0.097	0.381 ± 0.103	0.747 ± 0.045	4.181 ± 0.185	20.785 ± 0.990	0.408 ± 0.091
ICVF (Ghosh et al., 2023)	0.510 ± 0.002	0.334 ± 0.004	0.106 ± 0.002	0.276 ± 0.008	0.235 ± 0.003	0.677 ± 0.001	3.973 ± 0.003	11.029 ± 0.277	1.028 ± 0.013
DreamerV3 RSSM (Hafner et al., 2025)	0.768 ± 0.014	0.601 ± 0.013	0.288 ± 0.018	0.849 ± 0.016	0.314 ± 0.006	0.690 ± 0.005	5.006 ± 0.238	5.732 ± 0.141	0.656 ± 0.019
task02-keydoor3 (three rooms), 1600 episodes									
ACSI	0.789 ± 0.013	0.654 ± 0.010	0.282 ± 0.010	0.817 ± 0.012	0.670 ± 0.023	0.895 ± 0.007	3.741 ± 0.004	3.277 ± 0.039	0.128 ± 0.007
ACSI (no noise)	0.784 ± 0.015	0.622 ± 0.009	0.253 ± 0.007	0.818 ± 0.011	0.625 ± 0.034	0.885 ± 0.010	3.747 ± 0.002	3.412 ± 0.096	0.167 ± 0.009
Our target + actions	0.791 ± 0.012	0.614 ± 0.008	0.243 ± 0.007	0.836 ± 0.005	0.599 ± 0.023	0.880 ± 0.008	3.746 ± 0.004	3.331 ± 0.043	0.182 ± 0.003
Causal-state RNN (Zhang et al., 2019)	0.714 ± 0.007	0.600 ± 0.007	0.230 ± 0.005	0.768 ± 0.014	0.494 ± 0.009	0.836 ± 0.004	3.701 ± 0.003	13.267 ± 0.139	0.491 ± 0.011
CPC	0.677 ± 0.001	0.543 ± 0.008	0.195 ± 0.003	0.714 ± 0.007	0.452 ± 0.004	0.817 ± 0.002	3.706 ± 0.003	4.015 ± 0.098	0.082 ± 0.003
DBC	0.557 ± 0.011	0.377 ± 0.005	0.139 ± 0.002	0.129 ± 0.069	0.159 ± 0.005	0.546 ± 0.013	6.077 ± 0.088	15.896 ± 3.213	0.731 ± 0.049
MICo	0.524 ± 0.002	0.430 ± 0.005	0.148 ± 0.003	0.430 ± 0.002	0.223 ± 0.003	0.670 ± 0.003	4.226 ± 0.031	8.638 ± 0.521	1.142 ± 0.011
GRU d=512	0.813 ± 0.027	0.684 ± 0.017	0.325 ± 0.018	0.873 ± 0.029	0.565 ± 0.017	0.870 ± 0.006	3.742 ± 0.005	3.088 ± 0.114	0.294 ± 0.010
RePo (Zhu et al., 2023)	0.628 ± 0.003	0.575 ± 0.002	0.206 ± 0.003	0.642 ± 0.012	0.356 ± 0.006	0.774 ± 0.004	3.758 ± 0.004	7.690 ± 0.101	0.682 ± 0.005
with our occupancy target	0.720 ± 0.011	0.596 ± 0.008	0.239 ± 0.006	0.790 ± 0.009	0.516 ± 0.015	0.840 ± 0.005	4.647 ± 0.054	4.442 ± 0.305	0.411 ± 0.007
Self-predictive ϕ_L (Ni et al., 2024)	0.595 ± 0.004	0.558 ± 0.005	0.187 ± 0.002	0.498 ± 0.012	0.356 ± 0.007	0.765 ± 0.001	3.720 ± 0.002	23.132 ± 0.953	0.095 ± 0.007
ICVF (Ghosh et al., 2023)	0.511 ± 0.001	0.427 ± 0.004	0.137 ± 0.001	0.319 ± 0.004	0.257 ± 0.003	0.709 ± 0.001	3.817 ± 0.004	8.568 ± 0.182	0.935 ± 0.007
DreamerV3 RSSM (Hafner et al., 2025)	0.635 ± 0.004	0.576 ± 0.014	0.193 ± 0.009	0.715 ± 0.009	0.298 ± 0.014	0.705 ± 0.008	4.722 ± 0.105	6.863 ± 0.128	0.749 ± 0.013

The geometry behind the SPREAD. The minimality column of Table 1 is a number about how the visits of one true state are arranged, and Figure 6 names the structure the control keeps inside one of them, a split by elapsed time whose silhouette falls from 0.23 without the noise to 0.03 under ACSI, read off a 2-D t-SNE of the held-out representations of `task01-keydoor` under the seed-0 encoders of Figure 3.

The ICVF value fit and its failure to recover the state. ICVF is the one baseline whose failure could be mistaken for a training failure, and it is not, since across 6 seeds its objective falls from 0.10 to 0.007 and the learned $V(h, g, g)$ correlates with the measured first-arrival rate at $+0.36$ to $+0.38$, the range our own occupancy head reaches, while its representation stays at chance for the causal state at an AUC of 0.510. A low-rank bilinear value is satisfied as soon as ϕ keeps the projection that value depends on, which is direct evidence for the claim of Section 2 that a sufficient statistic has to carry the whole distribution of the future rather than one scalar functional of it.

Checkpoint selection and read-out. Two protocol choices are verified for neutrality in Table 9: the shared checkpoint criterion and the RSSM read-out. Selecting each baseline's checkpoint by its own validation loss instead of ours leaves it where it was or makes it worse, from 0.677 to 0.517 for the self-predictive ϕ_L and from 0.817 to 0.754 for CPC, and reading the RSSM baselines from the posterior mean rather than from the sample they define removes the handicap of that sampling noise and raises the DreamerV3 one-room AUC from 0.709 to 0.768, the best of these readings against our 0.994.

Reduction of the state by ignoring actions. Stepping on the centre cell of the mirror task secretly swaps the effect of the left and right actions, and the swap is invisible in the observation. Exact enumeration gives 389 interventional classes against 195 causal states out of 438 reachable configurations, which is what Proposition 2 predicts, the same enumeration giving 438 and 389 for the lamp task, whose bit is visible and therefore splits every class. Our encoder follows the smaller partition,

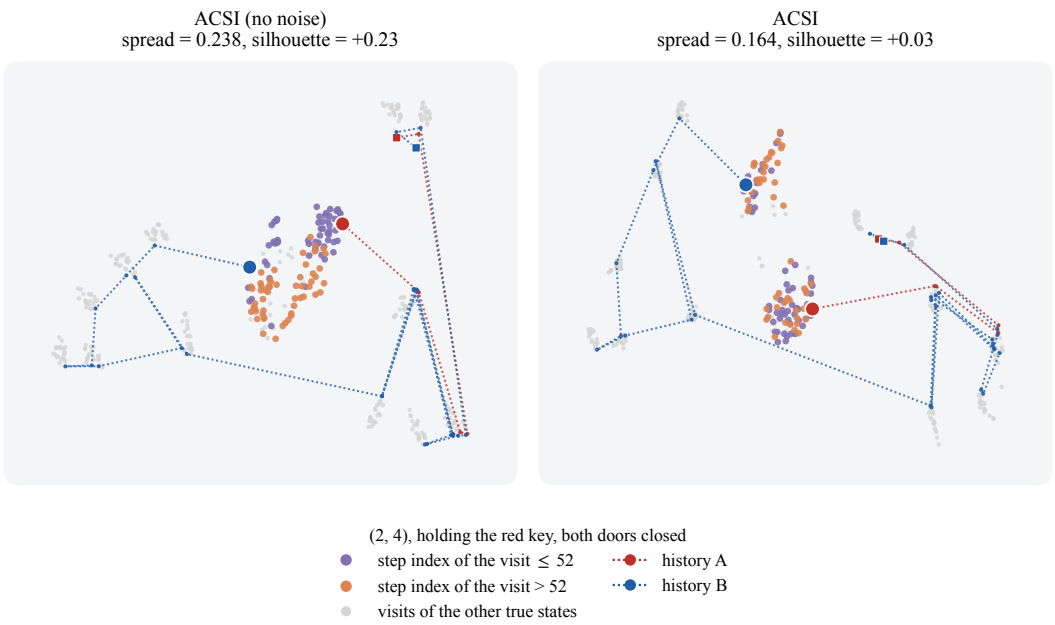


Figure 6: What the control still keeps inside one true state is elapsed time. The 114 visits of one true state are coloured by whether they fall at or before step 52 and every other true state is light grey, which extends the minimality paragraph of Section 4.

Table 9: Verification of two protocol choices for neutrality. The upper block shows that baselines selected by their own checkpoint criterion perform no better than under the shared one; the lower block shows that reading the RSSM baselines from the posterior mean rather than a posterior sample helps them, not the proposed method, which supports the fairness of Table 1.

	AUC		Probe		SPREAD	
Checkpoint criterion						
Method	ours	own	ours	own	ours	own
RePo	0.670 ± 0.004	0.697 ± 0.002	0.485 ± 0.008	0.497 ± 0.007	0.757 ± 0.007	0.674 ± 0.005
Self-predictive ϕ_L	0.677 ± 0.066	0.517 ± 0.001	0.460 ± 0.052	0.355 ± 0.006	0.408 ± 0.091	0.992 ± 0.005
ICVF	0.510 ± 0.002	0.510 ± 0.001	0.334 ± 0.004	0.334 ± 0.004	1.028 ± 0.013	1.011 ± 0.013
CPC	0.817 ± 0.015	0.754 ± 0.009	0.464 ± 0.015	0.419 ± 0.014	0.287 ± 0.010	0.204 ± 0.008
Read-out of the RSSM baselines						
Baseline	samp.	mean	samp.	mean	samp.	mean
RePo, task01-keydoor	0.670 ± 0.005	0.678 ± 0.005	0.484 ± 0.009	0.485 ± 0.008	0.758 ± 0.008	0.706 ± 0.008
RePo, task02-keydoor3	0.627 ± 0.004	0.628 ± 0.003	0.572 ± 0.002	0.575 ± 0.002	0.742 ± 0.009	0.682 ± 0.005
DreamerV3 RSSM, task01-keydoor	0.709 ± 0.013	0.768 ± 0.014	0.526 ± 0.017	0.601 ± 0.013	0.808 ± 0.008	0.656 ± 0.019
DreamerV3 RSSM, task02-keydoor3	0.612 ± 0.005	0.635 ± 0.004	0.522 ± 0.017	0.576 ± 0.014	0.860 ± 0.008	0.749 ± 0.013

since a probe reads the mirror bit at 0.544 against a majority rate of 0.543 whereas an encoder that reads actions stores it at 0.630 in 6 of 6 seeds, and under the recommended setting the probe stays at 0.542 while the SPREAD falls by 0.053.

E.2 DOWNSTREAM CONTROL

A state that recovers the causal partition is useful only insofar as a policy can exploit it, and this subsection extends Section 4 with every task and policy input, the aggregate statistics, the exact check of which tasks need memory, and the discrete read-out.

Downstream reinforcement learning in detail. The downstream claim holds where control needs memory and not elsewhere, which the runs show one condition at a time under the unified policy head of Appendix D. It survives the reporting of Agarwal et al. (2021) in Table 11, where in three rooms the noisy encoder solves 13 of 24 runs against 6 of 24 without the noise, at a probability of improvement of 0.66 with an interval of $[0.53, 0.79]$ that clears one half, while no other input on that

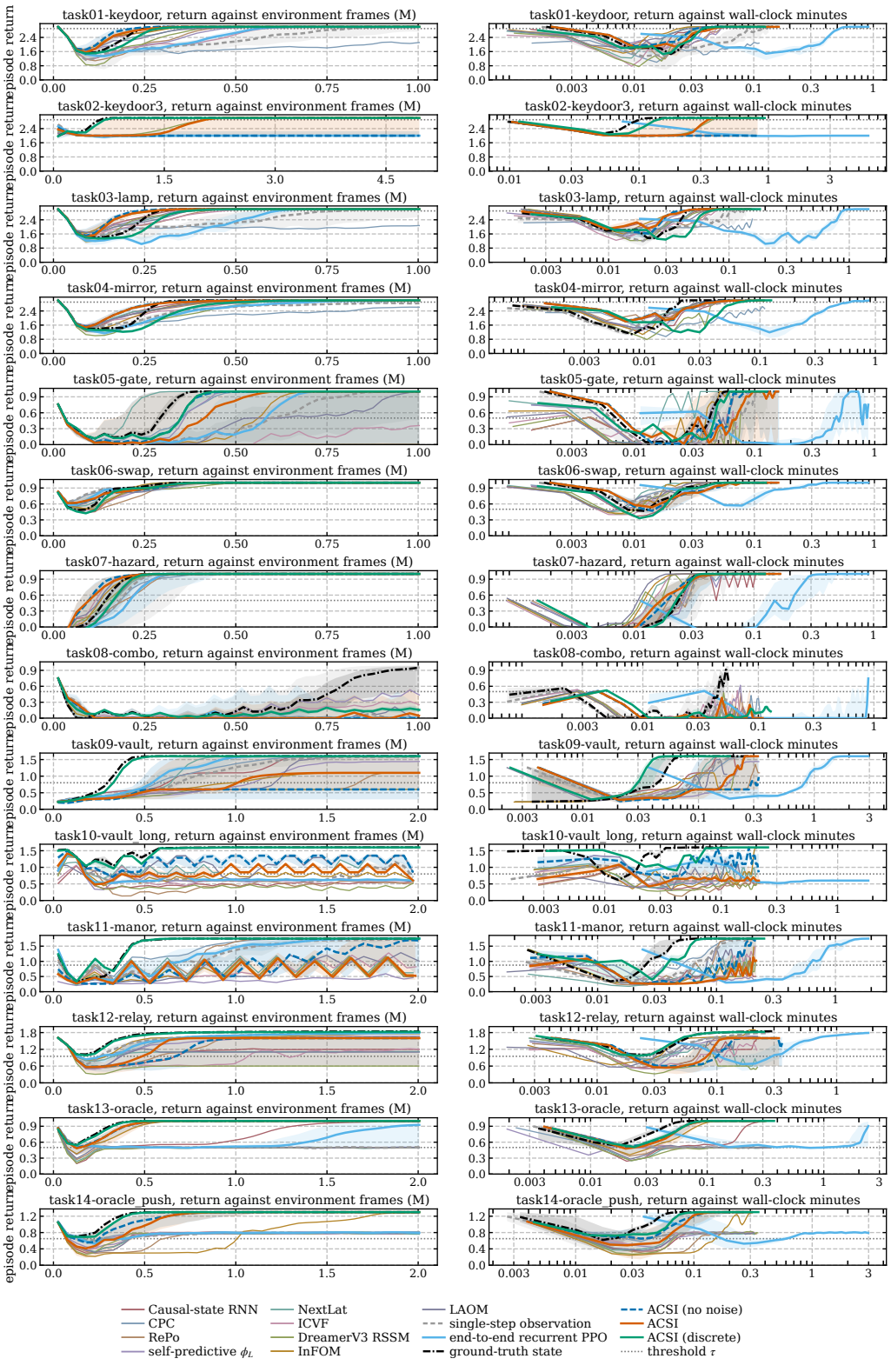


Figure 7: The learning curves behind every downstream number. One row per task gives the return against frames and against wall-clock minutes, each arm the median over its 24 runs against the threshold τ of that task, which is the evidence behind Tables 2 and 10.

Table 10: Downstream reinforcement learning per task and per policy input. Every (task, policy input) pair on which a PPO head was trained, at 24 runs each, which extends Section 4.

(a) RL IQM per task

Policy input	*t01-keydoor*	*t02-keydoor3*	*t03-lamp*	*t04-mirror*	*t05-gate*	*t06-swap*	*t07-hazard*	*t08-combo*	*t09-vault*	*t10-vault_long*	*t11-manor*	*t12-relay*	*t13-oracle*	*t14-oracle-push*
Causal-state RNN (Zhang et al., 2019)	3.00	2.00	3.00	3.00	0.18	1.00	1.00	0.00	1.51	0.60	0.81	1.19	0.86	0.83
CPC (van den Oord et al., 2018)	2.12	2.00	2.15	2.58	1.00	1.00	1.00	0.25	1.26	0.60	1.75	1.60	0.50	0.81
RePo (Zhu et al., 2023)	3.00	2.00	3.00	3.00	0.00	1.00	1.00	0.00	1.10	0.52	1.44	1.45	0.49	0.66
Self-predictive ϕ_L (Ni et al., 2024)	3.00	2.00	3.00	3.00	1.00	1.00	1.00	0.45	1.45	0.60	0.47	1.68	0.49	0.78
ICVF (Ghosh et al., 2023)	3.00	2.00	3.00	2.98	0.53	1.00	1.00	0.36	0.60	0.60	1.04	1.13	0.49	0.80
DreamerV3 RSSM (Hafner et al., 2025)	3.00	2.75	3.00	3.00	0.25	1.00	1.00	0.00	0.60	0.53	0.50	0.85	0.49	0.80
NextLat (Teoh et al., 2025)	3.00	2.00	3.00	3.00	1.00	1.00	1.00	0.00	1.46	0.60	0.87	1.13	0.49	0.66
InFOM (Zheng et al., 2026)	3.00	2.00	3.00	3.00	1.00	1.00	1.00	0.00	1.10	0.54	0.84	1.75	1.00	1.28
LAOM (Nikulin et al., 2025)	3.00	2.01	3.00	3.00	0.74	1.00	1.00	0.00	1.35	0.60	1.02	1.16	0.49	0.76
ACSI (no noise)	3.00	2.00	3.00	3.00	0.92	1.00	1.00	0.00	0.99	0.60	1.42	1.29	1.00	1.30
ACSI (prediction + adaptive noise)	3.00	2.58	3.00	3.00	0.88	1.00	1.00	0.04	1.10	0.60	0.82	1.29	1.00	1.30
ACSI (discrete, k=2$\lvert S^\star\rvert$)	3.00	3.00	3.00	3.00	1.00	1.00	1.00	0.20	1.60	1.60	1.75	1.83	1.00	1.30
ACSI (discrete, k=k_{95})	—	3.00	—	—	—	—	—	—	—	—	—	—	1.00	—

(b) frames@τ per task

Policy input	*t01-keydoor*	*t02-keydoor3*	*t03-lamp*	*t04-mirror*	*t05-gate*	*t06-swap*	*t07-hazard*	*t08-combo*	*t09-vault*	*t10-vault_long*	*t11-manor*	*t12-relay*	*t13-oracle*	*t14-oracle-push*
Causal-state RNN (Zhang et al., 2019)	0.30M	> 5.01M	0.31M	0.39M	> 1.00M	0.14M	0.16M	> 1.00M	0.89M	> 2.00M	> 2.00M	> 1.33M	0.59M	0.41M
CPC (van den Oord et al., 2018)	> 1.02M	> 5.01M	> 1.02M	> 1.02M	0.36M	0.04M	0.19M	> 1.00M	0.57M	> 2.00M	0.41M	0.39M	> 2.01M	0.33M
RePo (Zhu et al., 2023)	0.43M	> 5.01M	0.42M	0.43M	> 1.01M	0.16M	0.19M	> 1.00M	> 1.83M	> 2.00M	0.81M	0.69M	1.99M	0.65M
Self-predictive ϕ_L (Ni et al., 2024)	0.48M	> 5.01M	0.34M	0.43M	0.38M	0.12M	0.20M	> 1.00M	1.10M	> 2.00M	> 2.00M	0.45M	1.99M	0.57M
ICVF (Ghosh et al., 2023)	0.58M	> 5.01M	0.47M	0.71M	> 1.01M	0.12M	0.11M	> 1.00M	> 2.03M	> 2.00M	> 1.97M	> 1.54M	1.99M	0.33M
DreamerV3 RSSM (Hafner et al., 2025)	0.40M	1.91M	0.38M	0.56M	> 1.00M	0.10M	0.14M	> 1.00M	> 2.01M	> 2.00M	> 2.00M	> 1.97M	1.99M	0.37M
NextLat (Teoh et al., 2025)	0.35M	> 5.01M	0.37M	0.42M	0.20M	0.08M	0.17M	> 1.00M	0.67M	> 2.00M	> 2.00M	> 1.13M	> 2.03M	0.45M
InFOM (Zheng et al., 2026)	0.33M	> 5.01M	0.28M	0.44M	0.55M	0.12M	0.12M	> 1.00M	> 1.43M	> 2.00M	> 2.00M	0.33M	0.20M	1.02M
LAOM (Nikulin et al., 2025)	0.39M	> 5.01M	0.38M	0.41M	0.68M	0.08M	0.13M	> 1.00M	1.24M	> 2.00M	> 1.65M	> 1.18M	> 2.03M	0.45M
ACSI (no noise)	0.22M	> 5.01M	0.24M	0.31M	0.37M	0.06M	0.09M	> 1.00M	> 2.00M	> 2.00M	1.46M	0.69M	0.18M	0.24M
ACSI (prediction + adaptive noise)	0.26M	1.91M	0.26M	0.38M	0.43M	0.08M	0.08M	> 1.01M	> 1.49M	> 2.00M	> 2.00M	0.53M	0.20M	0.37M
ACSI (discrete, k=2$\lvert S^\star\rvert$)	0.28M	0.60M	0.37M	0.53M	0.37M	0.12M	0.16M	> 1.02M	0.33M	0.49M	0.37M	0.24M	0.08M	0.24M
ACSI (discrete, k=k_{95})	—	0.60M	—	—	—	—	—	—	—	—	—	—	0.08M	—

head reaches more than 4 of 24. In one room every state input already reaches the maximal return while the observation solves 16 of 24, so the only quantity with any resolution there is the number of frames to the threshold, where adaptive noise is not distinguishable from prediction alone.

Where control does not need memory the frozen state loses to the observation, and `task09-vault` is that case, so no downstream claim is made there. Under the same head the single-step observation, the ground-truth state and an end-to-end recurrent policy all solve 24 of 24 runs while every frozen state is behind them, from 18 of 24 for the self-predictive ϕ_L down to 11 of 24 for ACSI (no noise), and `task05-gate` reads the same way at 18, 17 and 24 of 24. The diagnosis is that the 128-dimensional state loses the position resolution this policy uses, as a linear read of the cell falls to 0.831 on policy while it stays at 1.000 on the two memory tasks, and concatenating the observation does not repair the deficit even when each block is normalized to unit RMS.

The restriction to two tasks is itself exact rather than empirical, since a stochastic memoryless policy already reaches 0.94 to 1.00 of $V^\star$ on seven of the ten levels that were checked exactly (Table 12).

Discrete states. Reading the same frozen state through a code book incurs no loss and helps through the input form, which Table 13 separates from the compression. Quantization on its own neither helps nor hurts, since the centroid rows match the continuous state, at 11 of 24 runs against 13 of 24 in three rooms, whereas the one-hot index at $k = 2|S^\star|$ solves 24 of 24 on both tasks, which on three rooms equals the ground-truth state. The code book that is needed is set by held-out purity rather than by the true state count, since the one-hot read-out at $k = k_{95}$ matches $k = 2|S^\star|$ exactly while $k = |S^\star|$ is slower in three rooms and short of the ceiling on the oracle task. The size at which purity saturates is itself a reading of how compact the state has become, 95% purity against the true minimal states taking 1.6 to 1.8 times as many codes, 512 against 280 in three rooms and 128 against 80 on the oracle task, so the code book size is itself a measurement of how compact the continuous state

Table 11: Downstream runs in the aggregate form of Agarwal et al. (2021). A run counts as solved when it reaches and holds $\tau = 2.9$ out of 3.0 and each probability of improvement is measured against the row marked as its reference, which extends Table 2.

Input	n	IQM [95% CI]	Solved [Wilson]	PoI vs. obs.	PoI vs. no noise
task01-keydoor (one room)					
observation	24	2.960 [2.825, 3.000]	16/24 [0.47, 0.82]	(reference)	0.27 [0.17, 0.38]
end-to-end recurrent PPO	8	3.000 [3.000, 3.000]	8/8 [0.68, 1.00]	0.73 [0.62, 0.83]	0.50 [0.50, 0.50]
frozen RePo state	8	3.000 [3.000, 3.000]	8/8 [0.68, 1.00]	0.73 [0.62, 0.83]	0.50 [0.50, 0.50]
frozen NextLat state	8	3.000 [3.000, 3.000]	8/8 [0.68, 1.00]	0.73 [0.62, 0.83]	0.50 [0.50, 0.50]
frozen CPC state	8	2.115 [1.775, 2.250]	0/8 [0.00, 0.32]	0.02 [0.00, 0.06]	0.00 [0.00, 0.00]
frozen self-predictive ϕ_L state	8	3.000 [3.000, 3.000]	8/8 [0.68, 1.00]	0.73 [0.62, 0.83]	0.50 [0.50, 0.50]
ACSI (no noise) state	24	3.000 [3.000, 3.000]	24/24 [0.86, 1.00]	0.73 [0.62, 0.83]	(reference)
ACSI state	24	3.000 [3.000, 3.000]	24/24 [0.86, 1.00]	0.73 [0.62, 0.83]	0.50 [0.50, 0.50]
observation $\oplus$ ACSI state	24	3.000 [3.000, 3.000]	24/24 [0.86, 1.00]	0.73 [0.62, 0.83]	0.50 [0.50, 0.50]
ground-truth state	24	3.000 [3.000, 3.000]	24/24 [0.86, 1.00]	0.73 [0.62, 0.83]	0.50 [0.50, 0.50]
task02-keydoor3 (three rooms)					
observation	24	2.000 [2.000, 2.002]	0/24 [0.00, 0.14]	(reference)	0.47 [0.34, 0.60]
end-to-end recurrent PPO	24	2.000 [2.000, 2.000]	0/24 [0.00, 0.14]	0.38 [0.30, 0.46]	0.39 [0.28, 0.50]
frozen RePo state	8	2.000 [2.000, 2.750]	2/8 [0.07, 0.59]	0.56 [0.40, 0.77]	0.53 [0.35, 0.71]
frozen NextLat state	8	2.000 [2.000, 2.000]	0/8 [0.00, 0.32]	0.42 [0.33, 0.48]	0.42 [0.31, 0.52]
frozen CPC state	8	2.000 [2.000, 2.000]	0/8 [0.00, 0.32]	0.42 [0.33, 0.48]	0.42 [0.31, 0.52]
frozen self-predictive ϕ_L state	8	2.000 [2.000, 2.000]	0/8 [0.00, 0.32]	0.42 [0.33, 0.48]	0.42 [0.31, 0.52]
ACSI (no noise) state	24	2.000 [2.000, 2.333]	6/24 [0.12, 0.45]	0.53 [0.40, 0.65]	(reference)
ACSI state	24	2.583 [2.167, 3.000]	13/24 [0.35, 0.72]	0.73 [0.61, 0.85]	0.66 [0.53, 0.79]
observation $\oplus$ ACSI state	24	2.167 [2.000, 2.583]	8/24 [0.18, 0.53]	0.59 [0.46, 0.72]	0.55 [0.42, 0.69]
ground-truth state	24	3.000 [3.000, 3.000]	24/24 [0.86, 1.00]	1.00 [1.00, 1.00]	0.88 [0.79, 0.96]

Table 12: Return of a memoryless policy on each Bunniescape level. This is why the downstream claim of Section 4 is restricted to the two tasks where memory is required, $V_{\text{mem-less}}$ being the best stochastic memoryless policy we found and hence a lower bound.

Task	$V^\star$	$V_{\text{mem-less}}$	ratio	$\text{success}_{\text{mem-less}}$
task05-gate	0.596	0.572	0.961	1.000
task06-swap	0.801	0.801	1.000	1.000
task07-hazard	0.710	0.706	0.994	1.000
task08-combo	0.467	0.456	0.978	1.000
task09-vault	0.947	0.889	0.939	1.000
task10-vault_long	0.681	0.467	0.686	0.000
task11-manor	0.901	0.843	0.936	1.000
task12-relay	1.009	1.006	0.997	1.000
task13-oracle	0.643	0.322	0.500	0.500
task14-oracle_push	0.758	0.506	0.667	0.500

has become. In frames the read-out reaches τ after 0.60M against the 1.91M of the continuous state in three rooms and after 0.08M against 0.20M on the oracle task, close to the ground-truth state's 0.50M and equal to its 0.08M.

Table 13: Compression and input form separated on the two memory tasks. Every row reads the same encoder and the same PPO head and only the input changes, which extends the read-out of Equation (8).

	`task02-keydoor3` (return ≤ 3, $\lvert\mathcal{S}^\star\rvert = 280$, $k_{95} = 512$)				`task13-oracle` (return ≤ 1, $\lvert\mathcal{S}^\star\rvert = 80$, $k_{95} = 128$)			
Policy input	k	final [95% CI]	solved	frames@τ	k	final [95% CI]	solved	frames@τ
Continuous ACSI state	*cont.*	2.542 [2.333, 2.750]	13/24	1.91M	*cont.*	0.940 [0.838, 1.000]	23/24	0.20M
Code centroid, $k = \lvert\mathcal{S}^\star\rvert$	280	2.417 [2.208, 2.625]	10/24	>5.01M	80	0.892 [0.801, 0.975]	21/24	0.20M
Code centroid, $k = 2\lvert\mathcal{S}^\star\rvert$	560	2.458 [2.250, 2.667]	11/24	>5.01M	160	1.000 [1.000, 1.000]	24/24	0.16M
One-hot code, $k = \lvert\mathcal{S}^\star\rvert$	280	3.000 [3.000, 3.000]	24/24	1.10M	80	0.914 [0.829, 0.981]	22/24	0.08M
One-hot code, $k = 2\lvert\mathcal{S}^\star\rvert$	560	3.000 [3.000, 3.000]	24/24	0.60M	160	1.000 [1.000, 1.000]	24/24	0.08M
One-hot code, $k = k_{95}$	512	3.000 [3.000, 3.000]	24/24	0.60M	128	1.000 [1.000, 1.000]	24/24	0.08M

E.3 REAL LASER DATA

Where no ground truth exists, the question moves from recovery to ranking, and this subsection extends Section 4 with every arm run on NCLT and a direct test of what reading actions would add.

Real laser data for every arm. Table 14 holds every arm of the capacity and noise sweep, the untrained geometric controls and the action-conditioned controls, under the four criteria of Appendix D. The selection rule picks the row we report, the $d = 256$ encoder at a fixed $\sigma = 0.03$ whose 512 codes are the largest count in Table 1, but it is not decisive against the $d = 128$ rows above it, so the advantage of the wider encoder shows on the query sessions rather than in the criterion. Against untrained geometric matchers the learned states win the ranking criteria and lose retrieval, since the best untrained arm reaches an aliasing AUC of 0.738 against 0.891 for ours while a 32-bin range histogram retrieves at recall@1 0.245 against 0.080, and retrieval falls further as the noise grows.

Table 14: Every arm run on the real laser data. The table extends Section 4 and the real-data block of Table 1.

Arm	Setting	aliasing AUC ↑	official AUC ↑	recall@1 ↑	cross-session AR@1 ↑	codes_{95}
NCLT, 5 sessions, 116672 frames: geometric descriptors, not trained						
36-beam log ranges (the tokenizer features)	36 log ranges	0.405	0.753	0.175	0.145	4
181-beam raw ranges	181 raw ranges	0.662	0.844	0.192	0.154	8
range histogram, 32 bins	32 range bins	0.738	0.792	0.245	0.210	4
NCLT, learned states, same words and budget						
ACSI (no noise)	K=135	0.758 ± 0.036	0.865 ± 0.006	0.078 ± 0.010	0.055 ± 0.006	53
+ fixed σ=0.03	K=135	0.711 ± 0.038	0.851 ± 0.005	0.061 ± 0.002	0.042 ± 0.001	53
ACSI (adaptive noise)	K=135	0.786 ± 0.026	0.846 ± 0.007	0.067 ± 0.002	0.047 ± 0.001	53
ACSI (no noise)	γ=0.95	0.872 ± 0.011	0.897 ± 0.003	0.086 ± 0.003	0.061 ± 0.003	53
+ fixed σ=0.03	γ=0.95	0.876 ± 0.002	0.901 ± 0.003	0.073 ± 0.001	0.050 ± 0.003	117
ACSI (adaptive noise)	γ=0.95	0.875 ± 0.004	0.894 ± 0.003	0.054 ± 0.002	0.039 ± 0.001	64
CPC	K=135	0.718 ± 0.012	0.850 ± 0.008	0.080 ± 0.002	0.056 ± 0.001	53
RePo	K=135	0.823 ± 0.017	0.903 ± 0.002	0.070 ± 0.002	0.061 ± 0.005	27
Self-predictive ϕ_L	K=135	0.704 ± 0.015	0.826 ± 0.021	0.089 ± 0.002	0.064 ± 0.002	365
DreamerV3 RSSM	K=135	0.787 ± 0.004	0.901 ± 0.006	0.034 ± 0.002	0.034 ± 0.001	27
ICVF	K=135	0.767 ± 0.013	0.902 ± 0.001	0.055 ± 0.001	0.044 ± 0.001	37
CPC	γ=0.95	0.743 ± 0.011	0.829 ± 0.004	0.084 ± 0.001	0.059 ± 0.001	8
RePo	γ=0.95	0.796 ± 0.003	0.893 ± 0.002	0.070 ± 0.001	0.055 ± 0.001	21
Self-predictive ϕ_L	γ=0.95	0.672 ± 0.034	0.828 ± 0.008	0.080 ± 0.004	0.058 ± 0.002	389
DreamerV3 RSSM	γ=0.95	0.794 ± 0.003	0.904 ± 0.003	0.038 ± 0.001	0.036 ± 0.001	43
ICVF	γ=0.95	0.764 ± 0.006	0.895 ± 0.002	0.053 ± 0.000	0.041 ± 0.001	21
Capacity and noise sweep on top of ACSI (no noise), same words and budget						
adaptive noise, $\rho^\star$=0.10	γ=0.95	0.874 ± 0.006	0.893 ± 0.003	0.053 ± 0.001	0.039 ± 0.001	53
adaptive noise, $\rho^\star$=0.05	γ=0.95	0.859 ± 0.004	0.890 ± 0.001	0.048 ± 0.001	0.036 ± 0.001	43
fixed σ=0.05	γ=0.95	0.880 ± 0.006	0.899 ± 0.003	0.068 ± 0.002	0.049 ± 0.001	53
fixed σ=0.10	γ=0.95	0.873 ± 0.005	0.896 ± 0.001	0.059 ± 0.001	0.043 ± 0.001	53
ACSI (d=256, fixed σ=0.03)	γ=0.95	0.891 ± 0.004	0.910 ± 0.002	0.080 ± 0.001	0.060 ± 0.001	512
d=256, fixed σ=0.03, 128 words	γ=0.95	0.850 ± 0.009	0.908 ± 0.001	0.085 ± 0.002	0.062 ± 0.002	75
NextLat	γ=0.95	0.814 ± 0.012	0.856 ± 0.001	0.057 ± 0.001	0.041 ± 0.001	27
InFOM	γ=0.95	0.851 ± 0.003	0.892 ± 0.003	0.098 ± 0.001	0.065 ± 0.001	85
LAOM	γ=0.95	0.752 ± 0.003	0.792 ± 0.002	0.047 ± 0.001	0.034 ± 0.000	213
NCLT with actions: 3 speed × 3 turn bins quantised from the logged poses; the action-free rows above are the matching controls						
ACSI (no noise)	γ=0.95	0.872 ± 0.011	0.897 ± 0.003	0.086 ± 0.003	0.061	53
+ actions	γ=0.95	0.891 ± 0.005	0.913 ± 0.002	0.103 ± 0.001	0.072	256
ACSI, d=256, fixed σ=0.03 (the **ACSI** row of Table 1)	γ=0.95	0.891 ± 0.004	0.910 ± 0.002	0.080 ± 0.001	0.060	512
ACSI + actions	γ=0.95	0.899 ± 0.020	0.923 ± 0.001	0.096 ± 0.000	0.071	683
Causal-state RNN (Zhang et al., 2019), one-step, actions (the † row of Table 1)	γ=0.95	0.800 ± 0.009	0.863 ± 0.002	0.064 ± 0.000	0.041	320
RePo, actions	γ=0.95	0.795 ± 0.011	0.906 ± 0.002	0.089 ± 0.002	0.066	27
DreamerV3 RSSM, actions	γ=0.95	0.792 ± 0.007	0.916 ± 0.004	0.055 ± 0.002	0.041	64

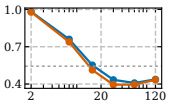
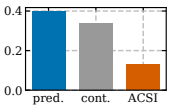
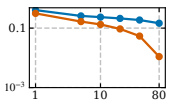
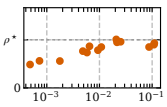
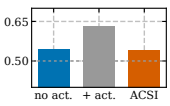
Actions on NCLT. Assumption 1 is the one assumption the real data can contradict, since the log is driven by a person, so we test Proposition 1 directly by quantizing the logged poses into nine actions and giving them to the same encoder under the same words, window, budget and 6 seeds (Table 14). Actions yield a small but consistent gain, the $d = 256$ row of Table 1 moving from 0.891, 0.910, 0.080 and 0.060 to 0.899, 0.923, 0.096 and 0.071 on the four criteria, so the action-free state is not equal to the action-conditioned one on this log, which is what Assumption 1 failing would predict. The loss is small against the spread between methods, since the action-conditioned baselines trained on the same nine actions reach an aliasing AUC of 0.795 and 0.792 against 0.891 without actions for us.

E.4 WHY IT WORKS: THE THEORETICAL PREDICTIONS

Having established that ACSI works, this subsection and the next examine why it does, extending Section 4 by turning each statement of Section 2 into a measurement and by reading the bound the state needs.

The five predictions as measurements. Each prediction of Section 2 is read inside a trained encoder rather than argued, and Table 15 states the measurement and the number that carries it for all five.

Table 15: The five predictions of Section 2 as measurements. Each row states one prediction, the measurement it rests on and the number that carries it, which is the evidence behind Section 4.

Prediction	Evidence	Key number	Task
Redundant memory that nothing maintains is forgotten by the contraction alone, with or without noise		probe $0.98 \to 0.44$ without noise ($0.98 \to 0.43$ with), majority 0.54	task12-relay
Redundancy the recurrence keeps rewriting survives more training and only noise removes it		within-state time R^2 0.398 (pred.) / 0.339 (continued) / 0.130 (ACSI)	task01-keydoor
A prediction-only recurrence does not contract geometrically, its $\hat{\rho}_K$ plateaus		$\hat{\rho}_K$ $0.403 \to 0.146$ over $K{=}1 \ldots 80$ without noise, $0.313 \to 0.011$ with	task12-relay
The controller holds $\hat{\rho}$ at $\rho^\star$ while the $\sigma^\star$ it needs spans two decades		$\sigma^\star \in [0.0005, 0.1099]$, $\hat{\rho} \in [0.073, 0.151]$ over 14 tasks	all tasks
Dropping the actions costs nothing except a bit that an action-reading encoder should not keep		AUC 0.993 action-free vs 0.993 with actions; mirror bit 0.544 vs 0.630 (majority 0.5)	task04-mirror

Escape from fixed noise by growth of the state. Without a bound on the state, fixed noise constrains nothing (Appendix B.5), and trained networks indeed grow their states under noise, the root-mean-square state going from 0.376 to 0.652 on a GRU and from 0.241 to 0.552 on an LSTM. We therefore bound the state with a layer norm or scale the noise with it, under which the two backbones respond identically at equal signal-to-noise ratio, so the difference between them comes from contraction and not from scale.

E.5 Why it works: design choices and sensitivity

With the mechanisms confirmed, what remains is how far they survive a change of component, of rate term, of backbone and of the one hyperparameter a user sets, which is the second half of Section 4.

Component and module ablations with the sensitivity sweeps. Nowhere but the recurrent channel is there a bottleneck that history must cross, so merging is a consequence of the recursion and not a generic effect of regularization, the noise moving the SPREAD by at most -0.010 in 4 of 6 seeds on the input embedding and in front of the prediction head against -0.090 in 6 of 6 on the recurrent state, which is also the only position that raises the true-state probe (Table 16).

Extra updates do not merge anything either, so the merging in Figure 8 is caused by the noise rather than by additional computation, since continuing phase 1 without noise raises the SPREAD by 0.028 in one room and by 0.011 in three rooms, fixed noise then lowers it by 0.025 and 0.049, and the controller lowers it by a further 0.076 in one room while raising the three-room probe by 0.039.

Noise applied before the memory exists destroys it, which is why long memories need the two-phase schedule. Training from scratch at $\sigma = 0.03$ on three-room KeyDoor costs 0.105 AUC and at the converged $\sigma^\star = 0.006$ still costs 0.022, the reverse direction of Proposition 4, while the two-phase controller gains 0.005 AUC, lowers the SPREAD by 0.039 in 5 of 6 seeds and raises the probe by 0.031 in 6 of 6. Read on the same reference in Table 16, it is the restart without noise rather than the noise itself that brings the three-room AUC gain, at $+0.027$ in 3 of 6 seeds against $+0.005$ in 4 of 6.

Smaller targets than the default merge further, down to 0.55 of the reference on `task09-vault` at $\rho^\star = 0.05$, but the same 0.05 drops one-room AUC from 0.994 to 0.910 (Figure 9a). A fixed schedule

is worse at the same job, since $\sigma_{\text{rel}} = 0.16$ drops three-room AUC to 0.719 and a fixed $\sigma = 0.1$, the level that merges one room and `task09-vault`, collapses three rooms in every one of the 6 seeds, whereas the controller lowers the three-room SPREAD from 0.167 to 0.128 at an AUC and a key probe at or above the ACSI (no noise) run. At 200 episodes ACSI lowers the one-room SPREAD only from 0.373 to 0.351 and costs AUC, from 0.923 to 0.893, whereas at 800 episodes it lowers it from 0.238 to 0.164 and at 2400 from 0.157 to 0.100, in both cases at an AUC of 0.994 or above (Figure 9c).

The target behaves differently on the capacity control at $d = 512$, where the controller leaves the three-room SPREAD at 0.292 against 0.294 for prediction alone, while a fixed $\sigma = 0.2$, the relative noise the $d = 128$ encoder carries, does merge and lowers it to 0.203 at a cost of AUC, from 0.828 to 0.777.

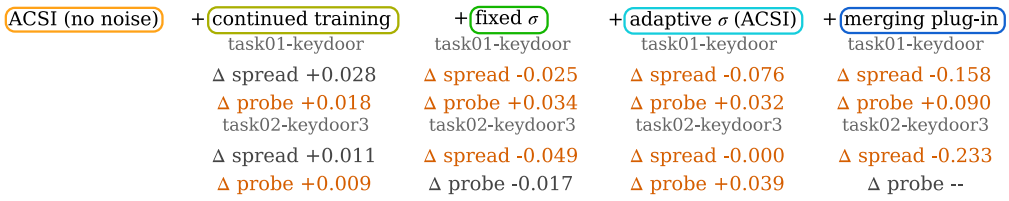


Figure 8: The method added one component at a time. Each step is the paired change of the SPREAD and of the true-state probe over 6 seeds, which extends the component reading of Section 4. The plug-in step sits on a transformer and is paired against a transformer of its own.

Table 16: Module ablations of ACSI, one module at a time. Each block is printed absolutely on its reference run and then as the paired change of every variant over 6 seeds, with the improving seeds in brackets and bold for at least 5 of 6.

Variant	Task	AUC ↑	Probe ↑	Value R^2 ↑	SPREAD ↓	Time R^2 ↓
Where the noise goes						
reference: ACSI (no noise)	task01-keydoor	0.993	0.803	0.888	0.238	0.661
input embedding	task01-keydoor	−0.001 (3/6)	**+0.016** (5/6)	**+0.011** (6/6)	−0.005 (4/6)	+0.004 (3/6)
before the prediction head	task01-keydoor	+0.001 (3/6)	−0.002 (3/6)	+0.003 (4/6)	−0.010 (4/6)	−0.007 (3/6)
recurrent state (ours)	task01-keydoor	−0.006 (2/6)	**+0.099** (6/6)	**+0.018** (5/6)	**−0.090** (6/6)	**−0.169** (6/6)
Two-phase training against training from scratch						
reference: ACSI (no noise)	task02-keydoor3	0.784	0.622	0.917	0.167	0.533
continued, no noise (control)	task02-keydoor3	+0.027 (3/6)	**+0.009** (5/6)	**+0.006** (6/6)	+0.011 (3/6)	+0.024 (0/6)
fixed σ=0.03 from scratch	task02-keydoor3	−0.105 (0/6)	−0.021 (2/6)	−0.081 (0/6)	**−0.086** (6/6)	**−0.136** (6/6)
fixed $\sigma^\star$=0.006 from scratch	task02-keydoor3	−0.022 (1/6)	−0.008 (1/6)	−0.016 (0/6)	**−0.038** (6/6)	**−0.044** (5/6)
two-phase, adaptive (ACSI)	task02-keydoor3	+0.005 (4/6)	**+0.031** (6/6)	**+0.015** (6/6)	**−0.039** (5/6)	−0.062 (4/6)
Merging plug-in on a transformer						
reference: ACSI (no noise)	task01-keydoor	0.835	0.308	0.432	0.482	0.700
consistency only (no prediction from $\hat{z}$)	task01-keydoor	−0.048 (0/6)	**+0.081** (6/6)	+0.031 (4/6)	**−0.069** (6/6)	**−0.154** (6/6)
stop-gradient on the target	task01-keydoor	−0.010 (3/6)	−0.037 (1/6)	+0.033 (4/6)	**−0.077** (6/6)	+0.087 (0/6)
full plug-in (ACSI)	task01-keydoor	−0.006 (2/6)	**+0.090** (6/6)	**+0.062** (5/6)	**−0.158** (6/6)	**−0.126** (6/6)

Backbones, each compared with its own prediction-only run

Backbone	Task	no noise (AUC / SPREAD)	+ fixed σ	+ adaptive noise	+ plug-in
GRU	task01-keydoor	0.993 / 0.238	−0.006 / **−0.090**	+0.001 / **−0.074**	−0.078 / **−0.150**
LSTM	task01-keydoor	0.890 / 0.197	**+0.032** / −0.003	+0.035 / +0.042	−0.001 / **−0.067**
Diagonal linear	task01-keydoor	0.676 / 0.801	**+0.002** / **−0.003**	**+0.069** / **−0.128**	**+0.079** / **−0.350**
Transformer	task01-keydoor	0.835 / 0.482	−0.004 / +0.008	−0.027 / +0.054	−0.006 / **−0.158**
GRU	task02-keydoor3	0.784 / 0.167	−0.091 / **−0.086**	+0.005 / **−0.039**	−0.096 / **−0.130**
LSTM	task02-keydoor3	0.667 / 0.124	+0.004 / +0.018	**+0.068** / +0.002	−0.008 / **−0.064**
Diagonal linear	task02-keydoor3	0.634 / 0.813	−0.004 / +0.005	**+0.038** / **−0.095**	**+0.037** / **−0.327**
Transformer	task02-keydoor3	0.694 / 0.363	**+0.020** / +0.003	+0.001 / +0.028	**+0.041** / **−0.233**

An explicit rate penalty instead of noise. No weight in the sweep of Table 17 improves on plain recurrent noise, so an explicit rate term either does nothing or removes everything, which is what Appendix B.5 predicts for a state whose rate is already capped. With a fixed σ the weight 10^{-2} drops the AUC to 0.721 and 10^{-3} drops the probe to 0.809, while the two smallest weights match the noise-only reference without merging more. With a learned σ the KL pushes σ to the scale of the prior and the two large weights collapse the encoder outright to an AUC of 0.520, and only 10^{-5} returns to 0.984, at a probe of 0.803 against the 0.902 of noise alone.

Backbones. The benefit of noise tracks how slowly a backbone contracts, which is what Theorem 1 implies, and where a backbone does not contract at all the plug-in supplies the channel. Measured as

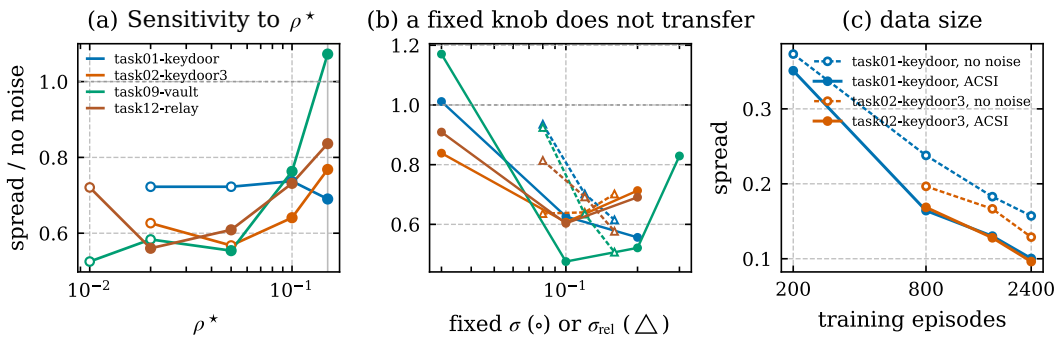


Figure 9: Sweeps of the single hyperparameter of ACSI and of the amount of training data. Panels (a) and (b) give the SPREAD against the ACSI (no noise) run of the same task as $\rho^\star$, a fixed σ and a fixed σ_{rel} vary, and panel (c) gives it against the number of training episodes. A filled marker in (a) keeps the AUC of that run within 0.01 and an open one does not.

Table 17: An explicit rate penalty in place of the recurrent noise. Every arm is trained in one room at 800 episodes over 6 seeds against plain recurrent noise at $\sigma = 0.03$.

	absolute					paired change against ACSI (no noise)		
Variant	AUC	Value	Probe	SPREAD	PR	AUC ↑	Probe ↑	SPREAD ↓
recurrent noise σ=0.03 (reference)	0.987 ± 0.004	0.907 ± 0.010	0.902 ± 0.004	0.148 ± 0.005	5.433 ± 0.153	−0.006 (2/6)	**+0.099** (6/6)	**−0.090** (6/6)
VIB, fixed σ, KL 10^{-2}	0.721 ± 0.014	0.298 ± 0.024	0.518 ± 0.013	0.276 ± 0.013	6.798 ± 0.142	−0.272 (0/6)	−0.284 (0/6)	+0.038 (0/6)
VIB, fixed σ, KL 10^{-3}	0.981 ± 0.004	0.822 ± 0.008	0.809 ± 0.011	0.230 ± 0.011	7.345 ± 0.208	−0.011 (0/6)	+0.006 (2/6)	−0.008 (3/6)
VIB, fixed σ, KL 10^{-4}	0.985 ± 0.006	0.879 ± 0.020	0.857 ± 0.022	0.172 ± 0.008	5.943 ± 0.169	−0.007 (1/6)	**+0.055** (5/6)	**−0.065** (6/6)
VIB, fixed σ, KL 10^{-5}	0.986 ± 0.006	0.902 ± 0.015	0.893 ± 0.007	0.150 ± 0.011	5.538 ± 0.130	−0.007 (1/6)	**+0.090** (6/6)	**−0.088** (6/6)
VIB, learned σ, KL 10^{-2}	0.520 ± 0.001	0.085 ± 0.003	0.327 ± 0.005	0.958 ± 0.003	2.519 ± 0.023	−0.473 (0/6)	−0.475 (0/6)	+0.720 (0/6)
VIB, learned σ, KL 10^{-3}	0.520 ± 0.002	0.091 ± 0.002	0.335 ± 0.005	0.930 ± 0.014	2.423 ± 0.065	−0.473 (0/6)	−0.467 (0/6)	+0.692 (0/6)
VIB, learned σ, KL 10^{-4}	0.826 ± 0.025	0.444 ± 0.034	0.606 ± 0.016	0.126 ± 0.012	5.906 ± 0.125	−0.166 (0/6)	−0.196 (0/6)	**−0.112** (6/6)
VIB, learned σ, KL 10^{-5}	0.984 ± 0.005	0.902 ± 0.017	0.803 ± 0.009	0.115 ± 0.008	6.151 ± 0.159	−0.009 (1/6)	+0.001 (2/6)	**−0.123** (6/6)

the residual of a relative perturbation after 100 steps, the GRU contracts slowly at 0.135 and noise speeds it up to 0.021 while merging states, the LSTM already contracts strongly at 0.019, the diagonal linear recurrence keeps slow modes that barely decay at 0.185, and the causal transformer has no recurrent channel at all. Against its own ACSI (no noise) run the controller therefore lowers the SPREAD by 0.039 on a GRU and by 0.095 on a diagonal linear recurrence, gains nothing on an LSTM at $+0.002$, and on a transformer the merging plug-in takes its place and gains 0.041 AUC with a SPREAD lower by 0.233, whereas on a GRU that plug-in duplicates an existing bottleneck and hurts. Where the constraint is infeasible the controller diverges, as Proposition 4 predicts, since the diagonal linear recurrence must keep its decay near 1 to remember the key and is driven to the noise ceiling while $\hat{\rho}_K$ stays at 0.39.

The plug-in with its cost and its design choices. The cost the plug-in carries is memory rather than accuracy, which the plug-in block of Table 16 isolates one design choice at a time. On a one-room transformer it lowers the recall of door openings by 0.20 to 0.28 while the key probe is unchanged at 0.849 against 0.847, cutting the gradient to the consistency target restores most of that recall and removes the merging, and stronger consistency noise merges more at unchanged accuracy, with the SPREAD falling by 0.109, 0.123 and 0.158 at noise 0.05, 0.1 and 0.3.

Sensitivity to the contraction target. The default $\rho^\star = 0.15$ is conservative rather than optimal, and the best value depends on how long a memory the task requires. Sweeping the target over 0.15, 0.10, 0.05, 0.02 and 0.01 at 6 seeds each, the best operating point subject to losing no more than 0.01 of sufficiency is 0.15 in one room and 0.10 in three rooms, 0.05 in the vault task and 0.02 in the relay task, where the SPREAD falls from 0.122 to 0.067 and from 0.123 to 0.069 while the probe rises. Setting KeyDoor to the same small targets causes it to fail, since at 0.05 in one room the AUC falls by 0.084 and the key probe by 0.087, which is the failure Corollary 1 predicts when the target leaves the feasible range of a task whose key must survive 70 steps, so we keep one default and leave it to be lowered by a practitioner who knows the memory span of the task.

F LIMITATIONS IN DETAIL

This section unpacks the limitations of Section 5. The theory assumes finitely many causal states over a finite alphabet, so continuous observations such as the laser ranges are quantized into words before any of it applies. The learned state is the causal state of the word process under the data policy, and Proposition 1 further requires that policy to depend on history only through the interventional class, so sufficiency for control is supported by our reinforcement learning results rather than by the theory. The target $\rho^\star$ is a single conservative default of 0.15 rather than a universally optimal value, since the best operating point in Appendix E.5 is 0.10 in three rooms and 0.02 in the relay task. The median residual the controller reads is a proxy for the pace of forgetting rather than the uniform contraction that part (ii) of Theorem 1 assumes, since a median over perturbation directions does not bound the slowest directions of the K-step map, whose largest singular value stays near one on all of our encoders.