Is Manipulation Hard for Successor Measures?

Aditya Mohan[1], **Caleb Chuck**[2], **Siddhant Agarwal**[2], **Amy Zhang**[2]
a.mohan@ai.uni-hannover.de

[1]Leibniz University Hannover
[2]University of Texas at Austin

Abstract

Many unsupervised reinforcement learning methods can be viewed as tractable approximations to policy-dependent successor measures. We study why these methods perform unevenly in the state-based OGBENCH `cube-single-play-v0` environment. Our hypothesis is that learned representations or scalar values may fail to distinguish nearby grasp states with different future occupancies. We decompose rollouts into approach, control acquisition, maintain control, and transport across five methods: GCIQL, CRL, FB, RLDP, and PSM. All methods approach the cube almost always, and most acquire control, but the lower-performing methods fail mainly during the maintain control and transport phases. State-support gaps broadly covary with performance: GCIQL shows almost no gap between successful and failed rollout states, whereas, for every lower-performing method, failed pickup and carry states are farther from the offline data. However, when initialized from in-hand states sampled directly from the offline dataset, RLDP and PSM rank grasp states near chance and FB only weakly by eventual task success, while CRL is intermediate, and GCIQL reaches a rank-AUC of 0.82. Linear probes further suggest that FB contains task-success information that is not fully preserved by its scalar readout, whereas RLDP contains less linearly accessible information. Together, these findings suggest that effective successor-measure approximations for manipulation must not only reach and lift the object, but also assign meaningfully different values to acquired-control states that support stable transport and those that lead to loss of control.

1 Introduction

Unsupervised reinforcement learning (URL) promises reusable control: an agent learns a policy, representation, or value structure without task rewards and later adapts it to downstream goals. This is especially appealing in robotics, where the same workspace may contain many possible goals, but collecting new interaction data for every reward is expensive. URL methods can be understood as learning tractable approximations to policy-dependent successor measures, differing in how they parameterize, compress, or extract information about future state occupancy (Agarwal et al., 2025a). Under this view, goal-conditioned values, contrastive state-goal values, explicit successor-measure factorizations, and predictive representations provide different approximations to future occupancy (Eysenbach et al., 2021; 2022; Ma et al., 2022; Touati and Ollivier, 2021; Touati et al., 2023).

In many control problems, a local interaction state can sharply change which future states remain reachable. An object may be stably held or slipping; a door may be latched or unlatched; and a robot foot may be in stance or swing. At the level of the full physical state, each of these systems remains Markovian and has a well-defined successor measure. The difficulty arises when the observation or learned representation fails to distinguish interaction states with different future occupancies. Two

states may have nearly identical object and robot poses, for example, while one corresponds to stable control and the other to an incipient slip.

In this work, we investigate whether contact-rich manipulation exposes a failure mode in successor-measure approximations: learned representations or values may preserve coarse geometric reachability while failing to distinguish interaction states with different future occupancies.

Hypothesis 1 (Interaction-state resolution)**.** *Successor-measure approximations can fail in manipulation when they assign similar representations or values to nearby states with different future occupancies. In cube manipulation, the relevant distinction is between states that support stable transport and states that lead to slipping or dropping.*

We formalize this discrepancy as a local successor gap: nearby states can have substantially different goal-relevant future occupancies, and therefore different probabilities of eventual task success.

We study this hypothesis in the state-based OGBENCH `cube-single-play-v0` environment (Park et al., 2025a), comparing five goal-conditioned, contrastive, predictive, and successor-measure methods. We decompose each rollout into four phases: approach, control acquisition, maintain control, and transport. This separates failures during approach from failures that occur after control has been acquired. Across methods, approach failures are rare, and most performance differences arise during maintain control and transport. State-support gaps broadly covary with performance: GCIQL shows almost no difference between successful and failed rollout states, whereas every lower-performing method shows positive gaps in pickup and carry. However, when initialized from in-hand grasp states sampled directly from the offline dataset, RLDP and PSM rank eventual task success near chance and FB only weakly, while CRL is intermediate. This suggests that initial-state support alone does not explain the observed value-ranking failures.

Our contributions are threefold: (i) We formulate interaction-state resolution as a potential failure mode of successor-measure approximations: nearby states can have different future occupancies while mapping to similar learned representations. (ii) We introduce a phase-based analysis of cube pick-and-place that separates approach, control acquisition, maintain control, and transport. (iii) We apply this analysis to five state-based methods on OGBENCH `cube-single-play-v0` and find that performance differences arise primarily after control acquisition. State-support gaps broadly covary with performance, while geometric-decoding and counterfactual value-ranking probes show that weak value ranking persists even for grasp states sampled from the offline dataset.

2 Preliminaries

We consider a reward-free Markov decision process $(\mathcal{S}, \mathcal{A}, P, \gamma)$ with state space $\mathcal{S}$, action space $\mathcal{A}$, transition kernel P, and discount factor γ. For a policy π, the successor measure from (s, a) to a measurable set $X \subseteq \mathcal{S}$ is:

$$M^\pi(s, a, X) = \mathbb{E}_\pi \left[\sum_{t \geq 0} \gamma^t \mathbf{1} s_{t+1} \in X \mid s_0 = s, a_0 = a \right]. \tag{1}$$

When the state space is discrete, we write $M^\pi(s, a, s^+)$ for the discounted occupancy of state s^+. For a state reward r, policy evaluation can be written as:

$$Q_r^\pi(s, a) = \int_{\mathcal{S}} M^\pi(s, a, ds^+) r(s^+). \tag{2}$$

This identity motivates successor measures for URL: a reward-independent representation of future occupancy can be combined with a downstream reward to evaluate task-specific policies (Agarwal et al., 2025a; Dayan, 1993; Touati and Ollivier, 2021; Touati et al., 2023).

Representing exact successor measures is generally impractical in large or continuous state spaces, particularly when the representation must cover many policies. Practical URL methods therefore

impose additional structure. For example, FB uses a low-rank approximation:

$$M^{\pi_z}(s, a, ds^+) \approx F(s, a, z)^\top B(s^+)\, \rho(ds^+), \tag{3}$$

where ρ is a reference state distribution and z indexes a reward or policy embedding (Touati and Ollivier, 2021; Touati et al., 2023). Other methods can be viewed as learning lower-dimensional or goal-specific projections of future occupancy (Agarwal et al., 2025a). Goal-conditioned methods learn values or policies indexed by a target, contrastive methods learn state–goal values, and predictive methods learn latent transition structure intended to support long-horizon control (Eysenbach et al., 2021; 2022; Ma et al., 2022). Each objective therefore determines which state distinctions are preserved for downstream control.

2.1 Successor-measure view of the evaluated methods

Let $o_G(s)$ denote the task-relevant goal coordinates extracted from state s. For a goal g, define the success set

$$G(g) = \left\{ s^+ : \left\| o_G(s^+) - g \right\| \leq \epsilon_G \right\}. \tag{4}$$

The corresponding goal-relevant successor mass is

$$m_g^\pi(s, a) = M^\pi(s, a, G(g)). \tag{5}$$

This quantity measures the discounted future occupancy of the goal set under π. Because repeated visits contribute additional mass, m_g^π is not in general equal to the probability of reaching the goal set. We use m_g^π only as a common reference: the evaluated methods do not all estimate it explicitly.

Under this view, GCIQL learns a goal-conditioned value $Q(s, a, g)$ (Park et al., 2025a), while CRL learns a bilinear state-action-goal value $f(s, a)^\top b(g)$ (Eysenbach et al., 2022). FB directly learns a low-rank factorization of policy-conditioned successor measures (Touati and Ollivier, 2021; Touati et al., 2023), and PSM learns a basis for representing a family of successor measures (Agarwal et al., 2025b). RLDP instead learns predictive features from which downstream task-conditioned policies are extracted (Jajoo et al., 2026).

Contact-rich manipulation can violate the assumption that geometrically similar states have similar futures. Nearby robot-object configurations may correspond to a stable grasp, an incipient slip, or a loss of control, and can therefore induce substantially different future occupancies. A useful learned representation or value must therefore preserve the interaction-state distinctions that affect future control, in addition to the coarse geometry of the robot, object, and goal.

3 Interaction-state resolution in successor measures

We now formalize the interaction-state hypothesis behind our analysis. Because the interaction state is part of the full physical state, the exact successor measure reflects its effect on future occupancy. Approximation error can arise when the physical state is compressed into an observation or learned representation. Let $\tilde{s} \in \widetilde{\mathcal{S}}$ denote the full physical state, and let $y = \Omega(\tilde{s})$ denote the state encoding used by the policy or value function, with $\iota(\tilde{s})$ denoting the local interaction state, such as whether an object is freely moving, stably held, or slipping. Here, Ω may denote either the observation map or a learned representation. Two full states can be nearby in representation space while having different interaction states and therefore different successor measures. The relevant question is which interaction-state distinctions are lost in the learned representation, and corresponding value-functions. We illustrate this distinction by describing how interaction state changes the transition dynamics.

3.1 Interaction-dependent transition dynamics

To illustrate how interaction state changes future occupancy, consider a binary projection $\omega(\tilde{s}) \in \{0, 1\}$ of the interaction state $\iota(\tilde{s})$. Here, $\omega = 0$ denotes a passive or pre-contact state, while $\omega = 1$

denotes a state in which the robot action directly affects the object. Let r_t denote the robot state and o_t the object state. For illustration, write the object transition as

$$o_{t+1} \sim \begin{cases} P_0(\cdot \mid o_t), & \omega_t = 0, \\ P_1(\cdot \mid r_t, o_t, a_t), & \omega_t = 1. \end{cases} \tag{6}$$

This binary projection records whether the robot action directly affects the object; finer interaction states may distinguish stable control, slipping, and loss of control. When $\omega_t = 0$, the object transition does not depend directly on the robot action. When $\omega_t = 1$, it depends on the robot state and action. Thus, similar robot–object configurations can have different successor measures when their interaction states induce different object dynamics.

3.2 Control acquisition and task success

In interaction-rich tasks, acquiring control of the object separates reaching it from completing the task. A policy may satisfy a coarse control-acquisition criterion while reaching a state from which retaining and transporting the object remains unlikely.

Let $C \subseteq \widetilde{\mathcal{S}}$ denote the control-acquisition set and define its first hitting time as $\tau_C = \inf\{t : \tilde{s}_t \in C\}$. In cube manipulation, states in C may correspond to either a stable grasp or an unstable grasp that is likely to slip. The event $\tau_C \leq H$ is therefore not sufficient for task success: success also depends on the state from which the policy begins transporting the cube.

Let $R \subseteq \widetilde{\mathcal{S}}$ denote the final-retention set and let $G_g \subseteq \widetilde{\mathcal{S}}$ denote the goal set for goal g. For a remaining horizon k, define the task-success value from state $\tilde{s}$:

$$v_k^\pi(\tilde{s}; g) = \Pr_\pi\left(\tilde{s}_k \in R \cap G_g \mid \tilde{s}_0 = \tilde{s}\right). \tag{7}$$

Conditioning on the first control-acquisition time gives

$$u_H^\pi(\tilde{s}, a; g) = \mathbb{E}_\pi\left[\mathbf{1}\{\tau_C \leq H\} v_{H-\tau_C}^\pi(\tilde{s}_{\tau_C}; g) \mid \tilde{s}_0 = \tilde{s}, a_0 = a\right]. \tag{8}$$

This decomposition separates whether the policy acquires control from whether it completes the task after control has been acquired.

Our experiments show that most performance differences arise after control acquisition. Policies generally reach and lift the cube, but many fail to retain it or transport it to the goal. Stable and unstable grasp states can both satisfy the same lifting criterion while inducing different successor measures. From a stable grasp, the policy's successor measure places more mass on retained-cube and goal states, whereas from an unstable grasp it places more mass on dropped-cube or repeated-pickup states.

3.3 Local successor gaps in learned values

For an observation scale $\epsilon > 0$ and remaining horizon k, define the local successor gap

$$\Gamma_{\epsilon,k}^\pi(g) = \sup_{\substack{\tilde{s}_1, \tilde{s}_2 \in C \\ \|\Omega(\tilde{s}_1) - \Omega(\tilde{s}_2)\| \leq \epsilon}} \left|v_k^\pi(\tilde{s}_1; g) - v_k^\pi(\tilde{s}_2; g)\right|. \tag{9}$$

A large $\Gamma_{\epsilon,k}^\pi(g)$ means that nearby states reached after lifting have substantially different task-success values. Such a gap arises when two states satisfy the same lifting criterion, but only one is likely to retain the cube and reach the goal.

Suppose a stable grasp state $\tilde{s}_s$ and an unstable grasp state $\tilde{s}_u$ produce the same state encoding,

$$\Omega(\tilde{s}_s) = \Omega(\tilde{s}_u) = y,$$

but have different task-success values:

$$v_s = v_k^\pi(\tilde{s}_s; g), \qquad v_u = v_k^\pi(\tilde{s}_u; g).$$

If p is the probability that encoding y corresponds to the stable grasp state, then the optimal encoding-based predictor under squared error is

$$\bar{v}(y) = pv_s + (1-p)v_u. \tag{10}$$

Its errors on the two states are

$$|v_s - \bar{v}| = (1-p)|v_s - v_u|, \qquad |v_u - \bar{v}| = p|v_s - v_u|. \tag{11}$$

Thus, if stable and unstable grasp states receive the same observation or representation, an encoding-based value estimate must underestimate the stable state, overestimate the unstable state, or both. Overestimating unstable grasp states can cause the actor to prefer actions that lift the cube but do not maintain control during transport.

This information loss may occur either in the learned representation or in the scalar readout. A bilinear value parameterization can express the distinction only if the representation encodes the relevant interaction-state information and the task-conditioned readout uses it. Our empirical probes therefore ask whether the learned representation contains information about task success from in-hand grasp states and whether the scalar value preserves that information.

3.4 Instantiation in cube manipulation

We study this hypothesis in the state-based OGBENCH `cube-single-play-v0` environment. Here, the relevant interaction state describes whether the cube is freely moving, stably held, slipping, or no longer controlled. The observation contains the gripper pose, cube pose, gripper opening, and contact indicators, but grasp stability is determined jointly by these variables rather than represented by a single explicit indicator. Consequently, nearby robot–cube configurations can differ in whether the policy will retain the cube during transport.

For goal g, we use the task-specific event

$$\varphi_g(\tilde{s}_{0:H}) = \mathbf{1}\{\tau_A \le H,\ \tau_C \le H,\ \tilde{s}_H \in R,\ \tilde{s}_H \in G_g\}, \tag{12}$$

where $\tau_A = \inf\{t : \tilde{s}_t \in A\}$, A is the reaching set, C is the control-acquisition set, R is the final-retention set, and G_g is the goal set. Under our operational definitions, acquiring control requires first reaching the cube, so

$$\tau_C < \infty \quad \Longrightarrow \quad \tau_A \le \tau_C. \tag{13}$$

Hence, whenever $\tau_C \le H$, the condition $\tau_A \le H$ is automatically satisfied. We retain it in φ_g only to make the phase sequence explicit. The corresponding task-success probability can therefore be written equivalently as

$$\begin{aligned} u_H^\pi(\tilde{s}, a; g) &= \Pr_\pi\left(\tau_A \le H,\ \tau_C \le H,\ \tilde{s}_H \in R \cap G_g \mid \tilde{s}_0 = \tilde{s}, a_0 = a\right) \\ &= \Pr_\pi\left(\tau_C \le H,\ \tilde{s}_H \in R \cap G_g \mid \tilde{s}_0 = \tilde{s}, a_0 = a\right). \end{aligned} \tag{14}$$

4 Diagnostic Method

We now turn the interaction-state hypothesis from Section 3 into a rollout-based analysis. The goal is to separate failures during approach and control acquisition from failures during maintain control and transport.

4.1 Evaluation setting

We study the state-based version of OGBENCH `cube-single-play-v0` over five held-out evaluation goals, denoted *task1* - *task5*. Using state observations removes visual encoding from the comparison and focuses the analysis on the learned values, representations, and policies. We evaluate five state-based URL methods: GCIQL, CRL, FB, RLDP, and PSM. We report interquartile mean (IQM) (Agarwal et al., 2021) success with stratified-bootstrap confidence intervals for aggregate performance. We then analyze deterministic closed-loop rollouts using the phase decomposition below.

4.2 Phase decomposition

We instantiate the sets from Section 3 using the simulator-state thresholds in Table 1. Each rollout is successful if all criteria are satisfied; otherwise, it is assigned to the first unmet criterion in the ordered sequence. This converts aggregate success into a phase-specific failure profile.

Table 1: Operational criteria for assigning the first unmet phase in state-based rollouts. The transport criterion is a diagnostic threshold (5 cm final cube-to-goal *position*, together with final retention) and does not reproduce the benchmark's own success signal, which uses a 4 cm position threshold and no retention requirement. Phase-defined success is therefore a diagnostic quantity and is not expected to match aggregate benchmark success exactly.

Phase	**Criterion**
Approach reached	Minimum gripper-to-cube distance < 6 cm at any point in the episode
Control acquired	Cube lifted > 3 cm above the table with the gripper closed for at least 5 consecutive steps
Maintain control	Final cube height above the table ≥ 3 cm
Transport (diagnostic)	Final cube *position* within 5 cm of the goal position

The criteria are computed directly from simulator states and applied identically to every policy. They do not depend on a method's learned value, representation, or model estimate. The same decomposition can therefore be applied to goal-conditioned, contrastive, predictive, successor-measure, and model-based methods. These thresholds are task-specific operational criteria: maintain control is a final-state criterion and does not require uninterrupted control throughout the remainder of the episode.

4.3 Cross-method failure profile

For each method, we report the percentage of episodes assigned to success, approach failure, control-acquisition failure, maintain-control failure, and transport failure. This distribution distinguishes methods that fail during approach or control acquisition from methods that acquire control but fail during maintain control or transport. We also report the post-acquisition share of failures:

$$\text{Post-acquisition share} = \frac{\text{Maintain fail} + \text{Transport fail}}{100 - \text{Success}}. \tag{15}$$

This quantity measures the fraction of failed episodes in which the control-acquisition criterion was satisfied before the rollout failed.

5 Results

Our experiments test the interaction-state hypothesis from Section 3. The evaluated methods rarely fail during approach; most performance differences arise after control has been acquired. We then

ask whether this pattern is explained by state-space support and whether learned values distinguish acquired-control states with different task outcomes. These diagnostics identify where policies fail and whether learned values preserve the interaction-state distinctions needed for control, but they are not designed to isolate a single architectural cause. Per-goal success, percentiles, and the full diagnostic tables are in Appendix A.

5.1 Aggregate success obscures where policies fail

Aggregate state-based success ranks the methods but says nothing about where they fail. Across 10 evaluation seeds, the overall inter-quartile mean (IQM) (Agarwal et al., 2021) success rates are GCIQL (71.8 ± 4.1), FB (52.0 ± 5.5), PSM (36.7 ± 4.5), CRL (25.8 ± 2.3), and RLDP (22.7 ± 6.0).[1]

The phase decomposition in Section 4 localizes where the methods fail (Figures 1 and 2). Every method almost always completes the approach phase, and most rollouts satisfy the control-acquisition criterion. The largest differences arise afterward, in maintain-control and transport failures.

FB and GCIQL differ little during approach and control acquisition. Relative to GCIQL, FB has 17.1 percentage points more maintain-control failures and 10.3 points more transport failures, partly offset by 3.2 points fewer control-acquisition failures. The same pattern appears for other methods: PSM and RLDP acquire control but fail to maintain it in 36% and 44% of episodes, respectively. Across the lower-performing methods, 65–83% of failed episodes occur after the control-acquisition criterion has been satisfied (Appendix A). Because the phase labels are computed only from simulator states, the same analysis can be applied to other methods without access to their learned values, representations, or models.

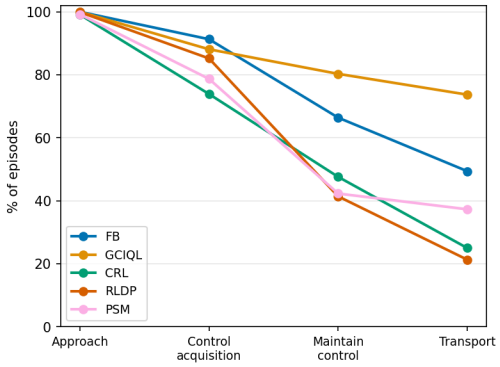

Figure 1: **Phase survival through cube manipulation.** Each curve shows the fraction of deterministic evaluation rollouts that satisfy each successive phase criterion: approach, control acquisition, maintain control, and transport.

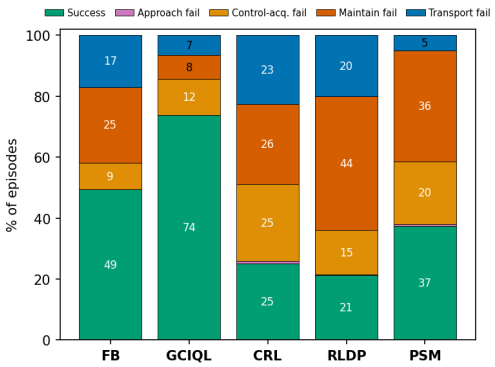

Figure 2: **Terminal outcomes under the phase decomposition.** Each rollout is assigned to success or to the first unmet phase criterion. Approach failures are negligible across methods, while most failures occur in maintain-control and transport.

5.2 State-support gaps covary with performance but do not explain value ranking

The support analysis groups individual states according to gripper and cube state. Because both the play data and policy rollouts contain repeated pick-and-place attempts, we use three state categories: *approach* (pre-contact), *pickup* (gripper closed, cube on the table), and *carry* (cube lifted and held).

[1] Aggregate success uses the environment's success signal under stochastic evaluation. The phase analysis uses deterministic rollouts and a separate diagnostic definition of success that additionally requires final retention and uses a 5 cm cube-to-goal position threshold (Table 1); its success percentage is therefore not expected to match aggregate benchmark success exactly.

These categories are distinct from the terminal failure phases: maintain-control and transport failures are assigned according to how the episode ends, whereas approach, pickup, and carry describe states encountered during the rollout.

One possible explanation is that failures after control acquisition occur in regions that are poorly represented in the offline data. Figure 3 reports the difference in nearest-neighbor distance to the offline dataset between failed and successful rollout states, grouped by state category and failure phase. Failed states are often farther from the data in the pickup and carry categories, especially for maintain-control failures (e.g. CRL pickup, $+1.83$).

These state-support gaps broadly covary with performance. GCIQL, the strongest method, shows almost no gap in the plotted pickup and carry categories, whereas every lower-performing method shows positive gaps. However, the gaps do not reproduce the exact performance ordering among the lower-performing methods, so state support is at most a partial explanation. The value-ranking probe below begins from in-hand states sampled directly from the offline dataset, so the initial states are supported by construction. Nevertheless, RLDP and PSM still rank eventual task success near chance, and FB only weakly. This shows that initial-state support alone does not explain the ranking failures, but it does not rule out limited action or transition support during the subsequent rollout. Weak value ranking may drive a policy toward poorly supported states, limited support may degrade the learned value or policy, or both effects may occur. Our diagnostics do not determine the direction of causality.

5.3 Learned values differ in how well they rank acquired-control states

Because initial-state support alone does not account for the ranking failures observed on states drawn from the dataset, we next investigate whether each learned value ranks in-hand grasp states by eventual success under the method's own policy. Coarse cube-to-goal geometry is not sufficient to explain performance. For methods with an explicit backward embedding, the embeddings learned by PSM and RLDP predict cube-to-goal distance substantially better than the embedding learned by FB (nonlinear R^2 of 0.90 and 0.75 versus 0.21). Nevertheless, both methods have higher maintain-control failure rates than FB (Appendix A). Encoding cube-to-goal geometry therefore does not guarantee that the policy will maintain control.

We evaluate value quality through simulator resets (Figure 4). We reset the simulator to 200 in-hand grasp states drawn from the offline data and pair each with the five held-out goals, giving $1{,}000$ state–goal pairs per agent seed. Because the flow policy is stochastic, we roll out $K{=}10$ independent continuations from each pair and label the pair a success when at least half of its continuations reach the goal. Each pair is evaluated independently, and the probe covers states from the offline dataset rather than only states generated by the learned policy. We then plot success rate as a function of the learned value percentile. We summarize ranking quality with the rank-AUC, i.e. the ROC-AUC of the learned value against this binary success label—the probability that a successful grasp receives a higher value than an unsuccessful one—computed within each goal over its 200 grasps and averaged across the five goals and the agent's training seeds (we use "rank-AUC" for this quantity throughout). Because the success labels are generated separately under each method's own policy, this probe measures whether a method's value is consistent with its own downstream outcomes rather than comparing all methods against a single shared target. GCIQL's value is the most predictive (rank-AUC 0.82): its top-value grasps succeed $\approx 99\%$ of the time, its bottom-value grasps $\approx 54\%$. For RLDP and PSM, the curves are nearly flat (rank-AUC 0.51 and 0.52), indicating that higher values do not correspond to higher success rates. FB shows a weaker relationship (rank-AUC approximately 0.64), while CRL is intermediate (0.73). Thus RLDP and PSM rank outcomes near chance, FB provides a weaker but positive ranking, and CRL is intermediate between FB and GCIQL; only GCIQL ranks acquired-control states reliably.

We next ask whether the weak ranking arises in the learned representation or in the scalar value derived from it. Using each agent's simulator-reset success labels, we fit a linear readout from the learned representation and compare its ranking with that of the learned value (Appendix A, Figure 5).

The probe suggests different bottlenecks for FB, RLDP, and PSM. FB is consistent with a readout bottleneck: a linear readout of its forward map reaches rank-AUC 0.83, while its learned scalar value reaches only 0.64. RLDP is consistent with a representation bottleneck under this probe: its linear representation readout reaches 0.68, below the raw-state baseline of 0.72, while its learned value reaches 0.51. We treat "representation bottleneck" cautiously, since a lower linear-probe score does not establish that the relevant information is absent—it may be nonlinearly encoded. PSM shows the same pattern more sharply: although its backward embedding resolves cube-to-goal geometry best of any method (nonlinear R^2 0.90; Table 5), a linear readout of the representation its value uses does not exceed the raw-state baseline for counterfactual success (0.66 versus 0.73), and the value itself ranks near chance (0.52). Resolving cube-to-goal geometry therefore does not imply that the representation encodes which grasps the policy will keep. For GCIQL, the learned value retains most of the linearly decodable signal available in its value features (readout 0.92, value 0.82), the strongest value ranking under this diagnostic.

These probes suggest that weak ranking can arise either in the learned representation or in the scalar readout, but they do not identify the readout form as the sole cause. GCIQL differs from FB, CRL, RLDP, and PSM in the learning objective, critic architecture, implicit regularization, and optimization stability, not only in its value readout. The evidence therefore supports a narrower conclusion: RLDP and PSM rank acquired-control states near chance and FB only weakly, whereas CRL and especially GCIQL rank them more reliably under this diagnostic. A matched readout ablation would be needed to attribute the difference specifically to bilinear versus MLP value form.

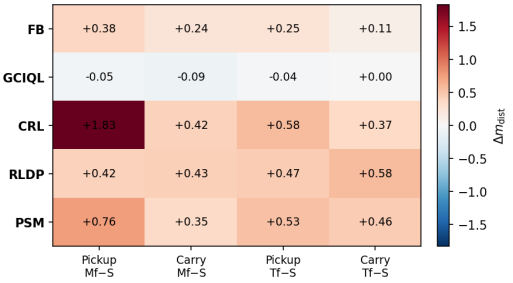

Figure 3: **Failure-success support gap.** Positive values indicate that failed rollout states are farther from the offline dataset under the nearest-neighbor metric. The gap is positive for every lower-performing method but is essentially zero for GCIQL, so state-support gaps broadly covary with performance.

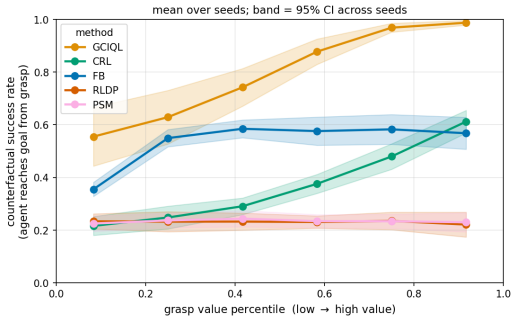

Figure 4: **Counterfactual value test.** For 200 in-hand grasps reset into the simulator, each paired with the five held-out goals (1,000 state–goal pairs) and labeled a success when at least half of its K=10 continuations reach the goal, success rate as a function of the grasp state–goal pair's within-goal value percentile. Exact rank-AUC values, computed per goal and averaged across goals, are reported in Table 6.

Takeaway. The phase decomposition changes the interpretation of state-based URL failures in cube manipulation. The evaluated methods usually complete approach and acquire control, but many failures occur afterward, in the maintain-control and transport phases. Failed pickup and carry states are farther from the offline dataset for the lower-performing methods but not for GCIQL, so state-support gaps broadly covary with performance. However, initial-state support alone does not explain why RLDP and PSM rank in-dataset grasp states near chance. Among the lower-performing methods, RLDP and PSM rank acquired-control states near chance and FB only weakly, while CRL is intermediate; only GCIQL ranks them reliably. Distinguishing such states is therefore a useful target for future offline URL objectives in manipulation.

6 Related Work

Unsupervised and goal-conditioned RL. URL methods learn policies, values, or representations in a reward-free or reward-agnostic phase and reuse them for downstream goals. Goal-conditioned methods learn values or policies indexed by target states (Eysenbach et al., 2021; Ma et al., 2022), while contrastive and metric-based methods learn goal-conditioned values or representations by distinguishing future states from negatives (Eysenbach et al., 2022; Park et al., 2024; Wang et al., 2023). Successor representations, successor features, and FB representations estimate future-state occupancy that can be combined with downstream rewards (Barreto et al., 2017; Dayan, 1993; Touati and Ollivier, 2021; Touati et al., 2023). Proto-value and proto-successor methods instead use learned bases or spectral structure to represent behavior or value spaces (Agarwal et al., 2025b; Farebrother et al., 2023; Mahadevan, 2005; Mahadevan and Maggioni, 2007). Recent work has argued that many URL objectives can be viewed as tractable approximations to successor measures (Agarwal et al., 2025a). Our work uses this view diagnostically: we ask whether learned values and representations distinguish interaction states that change future occupancy, rather than proposing a new URL objective.

Structured and controllable representations. Classical factored MDP work exploits state structure to make planning and learning more efficient (Boutilier et al., 1995; Guestrin et al., 2003; Kearns and Koller, 1999); see also Mohan et al. (2024). More recent controllable-representation methods aim to preserve action-relevant state information, often through inverse or multi-step inverse dynamics objectives (Efroni et al., 2022; Lamb et al., 2022; Levine et al., 2024; Rudolph et al., 2024). These works are closely related to our interaction-state hypothesis: the relevant abstraction should retain variables that change how actions affect future states. Our contribution is different in scope. We do not learn a new abstraction or assume a known factorization; instead, we diagnose whether learned values and representations retain outcome-relevant information in acquired-control states..

Contact-rich dynamics and manipulation. Contact-rich manipulation is a canonical setting where local interaction variables change effective dynamics. Classical work on pushing and manipulation emphasizes that contact mode, friction, and stability determine controllability and feasible motion (Lynch and Mason, 1996; Mason, 1986). Trajectory-optimization and contact-implicit methods explicitly model or optimize through contact transitions (Aceituno-Cabezas and Rodriguez, 2020; Jin, 2024; Manchester and Kuindersma, 2017; Patel et al., 2019; Posa et al., 2014). Learning-based contact models also study the difficulty of representing discontinuous or hybrid contact dynamics (Hogan and Rodriguez, 2016; Pfrommer et al., 2020). Our paper does not propose a contact model. Instead, we use contact-rich cube manipulation as a controlled diagnostic for a broader URL question: whether learned values and representations distinguish grasp states with different future occupancies.

Offline goal-conditioned benchmarks and diagnostics. Offline goal-conditioned RL evaluates whether policies or representations learned from static data can solve downstream goals without additional online collection (Kostrikov et al., 2022; Ma et al., 2022; Park et al., 2025a). Benchmarks such as OGBENCH report aggregate goal-reaching success, which is useful for comparing algorithms but does not identify the physical stage at which a policy fails (Park et al., 2025a). Our diagnostic decomposes manipulation rollouts into approach, control acquisition, maintained control, and transport. This separates approach and control acquisition from the ability to maintain control and transport the object.

7 Conclusion

We studied where state-based future-reachability methods fail in contact-rich cube manipulation. Our first step was to decompose each rollout into approach, control acquisition, maintained control, and transport. Across five methods, aggregate success hides a consistent failure profile: policies rarely fail to reach the cube and often acquire coarse control, but many failures occur *after* acquisition, when

the post-interaction tail must preserve control and reach the goal. Support diagnostics show that state-space coverage covaries with performance but does not fully explain it: sampling initial grasp states from the offline dataset shows that initial-state support alone is not a complete explanation, though it does not rule out limited action or transition support during the subsequent continuation. On these in-dataset states, RLDP and PSM still rank outcomes near chance and FB only weakly, while CRL is intermediate. Value-ranking diagnostics further suggest that the weakest methods' learned scores (RLDP, PSM, and to a lesser extent FB) only weakly distinguish in-hand post-acquisition states with different post-interaction futures. These results point to interaction-conditioned successor structure as a useful target for future offline goal-conditioned and reward-free control methods. The goal is not only to encode where the robot, object, and goal are, but also to preserve the interaction-mode distinctions that determine which tail successor regime follows a local interaction event.

References

B. Aceituno-Cabezas and A. Rodriguez. A global quasi-dynamic model for contact-trajectory optimization in manipulation. In *Robotics: Science and Systems*, 2020.

R. Agarwal, M. Schwarzer, P. Samuel Castro, A. C. Courville, and M. G. Bellemare. Deep reinforcement learning at the edge of the statistical precipice. In Ranzato et al. (2021).

S. Agarwal, C. Chuck, H. Sikchi, J. Hu, M. Rudolph, S. Niekum, P. Stone, and A. Zhang. A unified framework for unsupervised reinforcement learning algorithms. In *RLBrew: Ingredients for Developing Generalist Agents Workshop (RLC 2025)*, 2025a.

S. Agarwal, H. Sikchi, P. Stone, and A. Zhang. Proto successor measure: Representing the behavior space of an rl agent. In *Proceedings of the 42nd International Conference on Machine Learning (ICML'25)*, 2025b.

M. Bagatella, M. Pirotta, A. Touati, A. Lazaric, and A. Tirinzoni. TD-JEPA: Latent-predictive representations for zero-shot reinforcement learning. In *The Fourteenth International Conference on Learning Representations*, 2026.

A. Barreto, W. Dabney, R. Munos, J. J. Hunt, T. Schaul, H. P. van Hasselt, and D. Silver. Successor features for transfer in reinforcement learning. In I. Guyon, U. von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Proceedings of the 31st International Conference on Advances in Neural Information Processing Systems (NeurIPS'17)*. Curran Associates, 2017.

C. Boutilier, R. Dearden, and M. Goldszmidt. Exploiting structure in policy construction. In C. Mellish, editor, *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI'95)*. Morgan Kaufmann Publishers, 1995.

P. Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural Computation*, 1993.

Y. Efroni, D. Misra, A. Krishnamurthy, A. Agarwal, and J. Langford. Provably filtering exogenous distractors using multistep inverse dynamics. In *The Tenth International Conference on Learning Representations (ICLR'22)* ICL (2022). Published online: `iclr.cc`.

B. Eysenbach, R. Salakhutdinov, and S. Levine. C-learning: Learning to achieve goals via recursive classification. In *The Ninth International Conference on Learning Representations (ICLR'21)*. ICLR, 2021. Published online: `iclr.cc`.

B. Eysenbach, T. Zhang, S. Levine, and R. Salakhutdinov. Contrastive learning as goal-conditioned reinforcement learning. In Koyejo et al. (2022).

J. Farebrother, J. Greaves, R. Agarwal, C. L. Lan, R. Goroshin, P. S. Castro, and M. G. Bellemare. Proto-value networks: Scaling representation learning with auxiliary tasks. In *The Eleventh International Conference on Learning Representations (ICLR'23)* ICL (2023). Published online: iclr.cc.

C. Guestrin, D. Koller, R. Parr, and S. Venkataraman. Efficient solution algorithms for factored mdps. *Journal of Artificial Intelligence Research*, 2003.

F. Hogan and A. Rodriguez. Feedback control of the pusher-slider system: A story of hybrid and underactuated contact dynamics. *arXiv preprint arXiv:1611.08268*, 2016.

Proceedings of the International Conference on Learning Representations (ICLR'22), 2022. ICLR. Published online: iclr.cc.

Proceedings of the International Conference on Learning Representations (ICLR'23), 2023. ICLR. Published online: iclr.cc.

P. Jajoo, H. Sikchi, S. Agarwal, A. Zhang, S. Niekum, and M. White. Regularized latent dynamics prediction is a strong baseline for behavioral foundation models. In *The Fourteenth International Conference on Learning Representations*, 2026.

W. Jin. Complementarity-free multi-contact modeling and optimization for dexterous manipulation. *arXiv preprint arXiv:2408.07855*, 2024.

M. Kearns and D. Koller. Efficient reinforcement learning in factored mdps. In T. Dean, editor, *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI'99)*. Morgan Kaufmann Publishers, 1999.

I. Kostrikov, A. Nair, and S. Levine. Offline reinforcement learning with implicit q-learning. In *The Tenth International Conference on Learning Representations (ICLR'22)* ICL (2022). Published online: iclr.cc.

S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors. *Proceedings of the 36th International Conference on Advances in Neural Information Processing Systems (NeurIPS'22)*, 2022. Curran Associates.

A. Lamb, R. Islam, Y. Efroni, A. Didolkar, D. Misra, D. Foster, L. Molu, R. Chari, A. Krishnamurthy, and J. Langford. Guaranteed discovery of control-endogenous latent states with multi-step inverse models. *arXiv preprint arXiv:2207.08229*, 2022.

A. Levine, P. Stone, and A. Zhang. Multistep inverse is not all you need. *arXiv preprint arXiv:2403.11940*, 2024.

K. Lynch and M. Mason. Stable pushing: Mechanics, controllability, and planning. *The International Journal of Robotics Research*, 1996.

Y. J. Ma, J. Yan, Dinesh Jayaraman, and O. Bastani. Offline goal-conditioned reinforcement learning via f-advantage regression. In Koyejo et al. (2022).

S. Mahadevan. Proto-value functions: Developmental reinforcement learning. In L. Raedt and S. Wrobel, editors, *Proceedings of the 22nd International Conference on Machine Learning (ICML'05)*. Omnipress, 2005.

S. Mahadevan and M. Maggioni. Proto-value functions: A laplacian framework for learning representation and control in markov decision processes. *Journal of Machine Learning Research*, 8: 2169–2231, 2007.

Z. Manchester and S. Kuindersma. Variational contact-implicit trajectory optimization. In *International Symposium on Robotics Research*, 2017.

M. Mason. Mechanics and planning of manipulator pushing operations. *The International Journal of Robotics Research*, 1986.

A. Mohan, A. Zhang, and M. Lindauer. Structure in deep reinforcement learning: A survey and open problems. *Journal of Artificial Intelligence Research*, 79, 2024.

S. Park, T. Kreiman, and S. Levine. Foundation policies with hilbert representations. In R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning (ICML'24)*, volume 251 of *Proceedings of Machine Learning Research*. PMLR, 2024.

S. Park, K. Frans, B. Eysenbach, and S. Levine. Ogbench: Benchmarking offline goal-conditioned rl. In *The Thirteenth International Conference on Learning Representations (ICLR'25)*. ICLR, 2025a. Published online: iclr.cc.

S. Park, Q. Li, and S. Levine. Flow q-learning. In *Forty-second International Conference on Machine Learning*, 2025b.

A. Patel, S. Shield, S. Kazi, A. Johnson, and L. Biegler. Contact-implicit trajectory optimization using orthogonal collocation. *IEEE Robotics and Automation Letters*, 2019.

S. Pfrommer, M. Halm, and M. Posa. ContactNets: Learning discontinuous contact dynamics with smooth, implicit representations. In J. Kober, F. Ramos, and C. Tomlin, editors, *Proceedings of the 4th Annual Conference on Robot Learning (CoRL'20)*, volume 155. PMLR, 2020.

M. Posa, C. Cantu, and R. Tedrake. A direct method for trajectory optimization of rigid bodies through contact. *The International Journal of Robotics Research*, 2014.

M. Ranzato, A. Beygelzimer, K. Nguyen, P. Liang, J. Vaughan, and Y. Dauphin, editors. *Proceedings of the 35th International Conference on Advances in Neural Information Processing Systems (NeurIPS'21)*, 2021. Curran Associates.

M. Rudolph, C. Chuck, K. Black, M. Lvovsky, S. Niekum, and A. Zhang. Learning action-based representations using invariance. In *Reinforcement Learning Conference*, 2024.

A. Touati and Y. Ollivier. Learning one representation to optimize all rewards. In Ranzato et al. (2021).

A. Touati, J. Rapin, and Y. Ollivier. Does zero-shot reinforcement learning exist? In *The Eleventh International Conference on Learning Representations (ICLR'23)* ICL (2023). Published online: iclr.cc.

T. Wang, A. Torralba, P. Isola, and A. Zhang. Optimal goal-reaching reinforcement learning via quasimetric learning. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning (ICML'23)*, volume 202 of *Proceedings of Machine Learning Research*. PMLR, 2023.

A Additional results

This appendix provides the detailed numbers behind the main-paper analyses: per-goal success, the exact terminal-outcome distribution, the full state-support distances, the backward-embedding geometry probe, and the counterfactual value and representation-ranking analysis. These results support the main empirical chain: aggregate success hides post-acquisition failures; state-support gaps broadly covary with performance, while weak value ranking persists for initial states drawn from the offline dataset; coarse cube-to-goal geometry is insufficient; and learned values differ in how well they rank in-hand states by the agent's own eventual success. Throughout this subsection, *value* refers to the method-specific task-conditioned scalar used to rank states or actions.

A.1 Per-goal success

Table 2: State-based evaluation success by held-out goal (IQM $\pm$ half-width of the 95% stratified-bootstrap CI across 10 seeds). All methods use the Flow-Q learning with BC (FQL-BC) regularization (Bagatella et al., 2026; Park et al., 2025b) as the policy extraction mechanism (actor) and are evaluated at 1M steps.

Method	**task1**	**task2**	**task3**	**task4**	**task5**	**Overall**
FB	21.7 ± 10.8	73.3 ± 10.8	73.3 ± 6.7	36.7 ± 15.8	50.0 ± 13.3	52.0 ± 5.5
GCIQL	78.8 ± 8.0	74.8 ± 5.8	76.4 ± 7.4	67.2 ± 4.2	68.0 ± 5.2	71.8 ± 4.1
CRL	23.6 ± 3.2	26.8 ± 7.4	28.4 ± 5.6	23.6 ± 3.8	27.2 ± 5.4	25.8 ± 2.3
RLDP	28.3 ± 10.8	16.7 ± 6.7	35.0 ± 15.9	13.3 ± 12.5	18.3 ± 10.0	22.7 ± 6.0
PSM	50.0 ± 11.7	38.3 ± 16.7	73.3 ± 12.5	1.7 ± 4.2	11.7 ± 9.2	36.7 ± 4.5

A.2 Terminal-outcome distribution

Table 3: Cross-method terminal-outcome distribution (% of episodes, mean across 10 seeds; the first five columns sum to 100%). The post-acquisition share is $(\text{maintain} + \text{transport})/(100 - \text{success})$.

Method	**Success**	**Approach**	**Control acq.**	**Maintain**	**Transport**	**Post-acq. share**
FB	49.4	0.0	8.7	24.9	17.0	82.8%
GCIQL	73.7	0.0	11.9	7.8	6.7	54.9%
CRL	25.0	0.9	25.2	26.3	22.6	65.2%
RLDP	21.3	0.1	14.7	43.8	20.1	81.2%
PSM	37.3	0.8	20.5	36.4	5.0	66.0%

A.3 State-support analysis

Table 4: Nearest-neighbor distance m_{dist} from rollout states to states in the offline dataset (mean across seeds; larger values indicate lower support), grouped by state category and terminal outcome. S = success, Mf = maintain-control failure, Tf = transport failure. All methods use the FQL-BC and are evaluated at 1M steps under the same probe (20 episodes per goal per seed, across 5 goals and 10 seeds). Figure 3 plots the Mf−S and Tf−S differences for the pickup and carry categories.

Method	**Category**	**S**	**Mf**	**Tf**
GCIQL	approach	2.53	2.29	2.43
	pickup	3.05	3.00	3.01
	carry	2.67	2.58	2.67
FB	approach	2.27	2.38	2.37
	pickup	2.36	2.74	2.61
	carry	2.09	2.33	2.20
PSM	approach	2.17	2.23	2.44
	pickup	2.22	2.98	2.75
	carry	2.10	2.45	2.56
CRL	approach	2.50	3.02	2.71
	pickup	2.73	4.56	3.31
	carry	2.58	3.00	2.95
RLDP	approach	2.32	2.36	2.55
	pickup	2.64	3.06	3.11
	carry	2.22	2.65	2.80

A.4 Backward-embedding geometry probe

Table 5: Carry-regime resolution of the backward embedding B for cube-to-goal distance d, under a linear and a nonlinear (MLP) readout, with the maintain-control failure rate. This probe is restricted to methods with an explicit backward embedding. Better cube-to-goal geometric resolution in B does not imply a lower maintain-control failure rate.

Method	$R^2(B \to d)$ **linear**	$R^2(B \to d)$ **MLP**	**Maintain-control fail (%)**
FB	0.20	0.21	24.9
RLDP	0.55	0.75	43.8
PSM	0.90	**0.90**	36.4

A.5 Counterfactual value and representation ranking

The main value-ranking analysis uses an on-policy counterfactual label. We reset the simulator to in-hand grasp states, roll out each agent's own policy, and each state–goal pair is labeled positive when at least five of its ten continuations succeed. This directly tests whether the learned value ranks acquired-control states by eventual task success under the corresponding policy. Because the labels are generated separately under each method's own policy, the analysis measures consistency with each method's own downstream outcomes rather than comparison against a single shared target.

Table 6: On-policy counterfactual value test. We reset 200 in-hand grasp states in the simulator and pair each with the five held-out goals (1,000 state–goal pairs), rolling out each agent's own (flow) policy for $K=10$ independent continuations per pair (the flow policy is stochastic, so a single rollout is a noisy draw) and labeling a pair a success when at least half of its continuations reach the goal. We report the learned value's rank-AUC (the ROC-AUC of the value against this binary label), computed within each goal and averaged across goals and the method's flow-actor training seeds (9–10 seeds), together with the mean counterfactual success rate (the average fraction of successful continuations).

Method	**Counterfactual rank-AUC**	**Counterfactual success rate**
GCIQL	0.82	0.79
CRL	0.73	0.37
FB	0.64	0.54
RLDP	0.51	0.23
PSM	0.52	0.23

Figure 5 compares the learned value with a supervised linear readout from the representation used to compute it. This asks whether the representation contains linearly accessible information about the agent's own counterfactual success and whether the learned value preserves that information in its ranking.

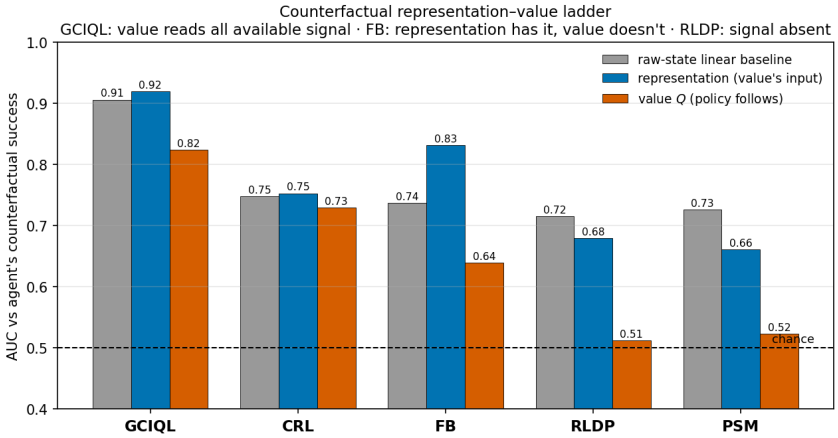

Figure 5: Counterfactual representation–value ladder. For each method, the probe uses the agent's own counterfactual success as the label. We compare a raw-state linear baseline, the best linear readout of the representation used by the scalar value (GCIQL penultimate value features; CRL $\phi(s, \pi)$; FB/RLDP/PSM the successor-feature map $F(s, \pi, z)$, which for PSM is $\psi(s, \pi, z)$), and the learned value's own ranking. GCIQL retains most of the linearly decodable signal available in its value features; FB contains more signal in F than is reflected in the learned scalar value; RLDP contains weaker signal under this probe; PSM resolves cube-to-goal geometry best (Table 5) yet its representation readout does not exceed the raw baseline and its value ranks near chance. The raw baseline is per-agent because counterfactual success depends on each policy, so the cleanest comparison is within method.