DO BETTER GOAL REPRESENTATIONS IMPROVE GOAL-CONDITIONED REINFORCEMENT LEARNING?

Anonymous authors
Paper under double-blind review

ABSTRACT

Goal-conditioned reinforcement learning (GCRL) relies heavily on how target goals are represented to the policy. While recent methods encode goals via temporal distance, occupancy, or controllability, it remains unclear how much downstream performance actually depends on representation quality. We study this in offline GCRL by constructing an exact temporal-distance goal representation in deterministic mazes. We then systematically corrupt its geometric quality while keeping the downstream learner fixed. Across OGBench navigation tasks and two algorithms, large changes in goal-representation quality produce almost no change in performance. However, applying the same interventions to the agent's current state more than doubles success, revealing the state pathway as the true bottleneck. Building on this insight, we show that simple random Fourier positional encodings substantially improve performance on the hardest navigation tasks without map information or objective modifications. Overall, our findings suggest that in state-based offline navigation, improving how the agent's current state is represented matters far more than refining the goal representation.

1 INTRODUCTION

Goal-conditioned reinforcement learning (GCRL) learns a single policy for reaching diverse target states by conditioning behaviour on a specified goal (Kaelbling, 1993; Schaul et al., 2015). Offline GCRL learns such policies from a fixed dataset, often using hindsight relabelling to construct goal-conditioned training examples (Andrychowicz et al., 2017; Park et al., 2025). A central question in this setting is how the goal should be represented for downstream control. Prior work has therefore developed representations intended to capture task-relevant relational structure beyond the raw goal observation.

Recent approaches characterise goals through future occupancy, temporal distance, or controllability, including contrastive representations (Eysenbach et al., 2022), value-implicit pre-training (Ma et al., 2023), quasimetric and temporal-distance embeddings (Wang et al., 2023; Park et al., 2024b; Myers et al., 2025b), and dual goal representations (Park et al., 2026a). Despite differences in their objectives, these methods are motivated by a common premise: a goal representation that more accurately captures reachability structure, particularly the temporal relationships between states, should enable more effective goal-conditioned control.

However, existing evaluations do not isolate the effect of goal-representation quality on downstream control. Prior comparisons typically vary the representation-learning procedure and the resulting embedding at the same time. This makes it difficult to determine whether performance differences arise from the information encoded by the representation or from other aspects of its learning process. We therefore ask a more direct question:

If a goal-conditioned agent were handed a perfect goal representation, how much better would it act?

Dual Goal Representations (DGR) provides such a setting by characterising goals through temporal distances and establishing their sufficiency for optimal control (Park et al., 2026a). In deterministic maze environments, these temporal distances correspond to shortest-path distances in the underlying transition graph and can therefore be computed directly. This allows us to construct the DGR representation from exact distances, which we term the ideal representation, and replace its learned

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.

approximation while leaving the downstream algorithm, training objective, and all other components unchanged.

We then systematically vary the quality of this representation to determine whether downstream control depends on the reachability information it encodes. The resulting analysis shows that learned goal-side representations achieve success rates comparable to the exact ideal representation. We therefore hypothesise that the state pathway, through which the agent's current state is presented to the policy, is the more critical bottleneck. The remainder of the paper focuses on developing a matched set of state-side interventions to test this hypothesis. Figure 1 contrasts the two sides of the interface.

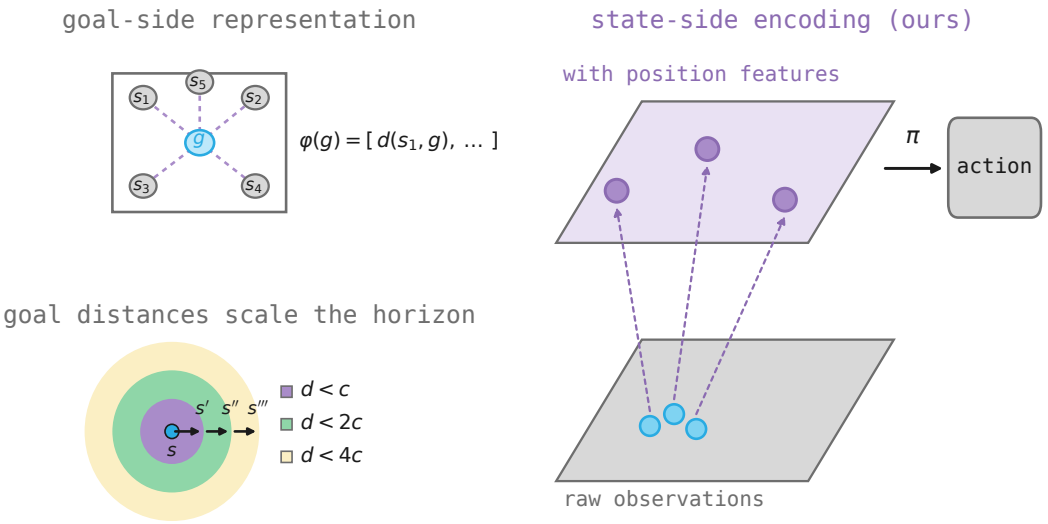


Figure 1: **Two sides of the same interface.** *Left, top:* goal-side representations, the axis this literature optimises. A goal is described by its temporal distances to other states. *Left, bottom:* the property that motivates them, namely that such distances compose, so that a policy accurate within one radius extends to two and four (Myers et al., 2025a). *Right:* our proposal. We encode the agent's own position instead, lifting observations that are close together in raw coordinates into a representation in which they are far apart, and from which the policy acts.

We propose a minimal method where random Fourier features of the agent's coordinates are appended to the observation. The method requires no maze layout and no modification to the learning objective. It only requires identifying which observation dimensions correspond to the agent's position. Despite its simplicity, it raises GCIVL success on antmaze-large from 32 to 84 and it roughly doubles success on humanoidmaze-medium (Figure 2). We show that with the downstream algorithm held fixed, the input pathway matters considerably more than the goal representation.

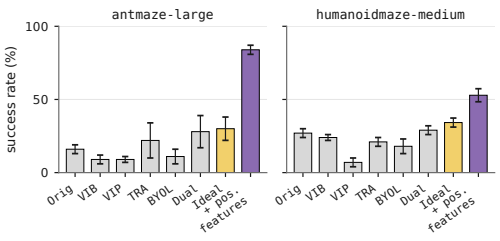


Figure 2: **Goal representations are saturated; the state pathway is not.** GCIVL on antmaze-large. *Grey:* published goal methods. *Yellow:* exact BFS goal representation. *Violet:* adding unprivileged state position features.

Our main **contributions** are threefold. First, we give the first measurement of an exactly computed ideal goal representation on continuous control, together with a probe-free way to measure representation quality, and show that representation quality is not the binding constraint on these tasks. Second, we identify the state pathway, rather than the goal representation, as the bottleneck, and support this with a matched pair of experiments in which the same intervention does nothing on the goal side and doubles performance on the state side. Third, we propose position features, a

simple method that requires no privileged information and substantially improves performance on the hardest navigation tasks we study.

2 PRELIMINARIES

Offline goal-conditioned reinforcement learning. We consider a controlled Markov process with state space $\mathcal{S}$, action space $\mathcal{A}$, and transition kernel $p(s' \mid s, a)$, together with a goal space $\mathcal{G} \subseteq \mathcal{S}$. A goal-conditioned policy $\pi(a \mid s, g)$ is trained to reach g from s. Following the benchmark convention (Park et al., 2025), the agent receives a reward of 0 on reaching the goal and -1 otherwise, discounted by γ, so that maximising return is equivalent to minimising the number of steps taken. Learning is offline: the agent has a fixed dataset $\mathcal{D}$ of trajectories collected by an unknown behaviour policy and may not interact with the environment during training. Goals are supplied by hindsight relabelling of states in $\mathcal{D}$ (Andrychowicz et al., 2017).

Temporal distances and dual goal representations. A natural way to structure goal representations is through reachability, quantified by temporal distance. Formally, let $d(s, g)$ denote the minimal expected time (or shortest-path cost) required for an agent to transition from state s to goal g under the environment dynamics. Dual Goal Representations (DGR; Park et al., 2026a) leverage this metric directly, representing a goal not by its raw visual or coordinate features, but by the collection of temporal distances from all states to that goal. The idealised representation is the function

$$\varphi^{\vee}(g) \;:=\; \big(s \mapsto d(s, g)\big),$$

which is provably sufficient for optimal control and invariant to dynamics that do not affect reachability. Because $\varphi^{\vee}(g)$ is infinite-dimensional, practical implementations instantiate it by evaluating distances against a finite set of K landmark states $a_1, \ldots, a_K$ sampled from the dataset:

$$\varphi(g) \;=\; \big[\, d(a_1, g),\; \ldots,\; d(a_K, g) \,\big] \in \mathbb{R}^K. \tag{1}$$

In standard DGR, the entries of equation 1 cannot be accessed directly in continuous domains and are instead approximated via a learned bilinear value parameterisation trained offline, which is then fed into downstream policy learners with a stop-gradient. Throughout this paper, we refer to this parametric estimate as the **learned** representation, and contrast it with the **ideal** representation obtained by evaluating equation 1 using exact ground-truth distances.

The goal interface. Any goal-conditioned network combines two arguments before producing an output. We make that combination explicit and call it the goal interface,

$$I : \mathcal{S} \times \mathbb{R}^K \to \mathbb{R}^m, \qquad \pi(a \mid s, g) \;=\; h\big(I(s, \varphi(g))\big),$$

where h is the action head, and likewise for value networks. Every goal-representation method we are aware of, including all of those we compare against, fixes the interface to concatenation followed by a multilayer perceptron,

$$I_{\text{concat}}(s, \varphi) \;=\; \text{MLP}\big([\, s\,;\, \varphi \,]\big). \tag{2}$$

The choice in equation 2 is inherited rather than argued for. The closest existing comparisons vary something else: Park et al. (2026a) compare aggregation functions inside the representation-learning objective, and compare early against late fusion for pixel encoders. Neither varies how a state-based goal representation is combined with the state inside the downstream policy. The representation φ, the interface I, and the state pathway are separate design axes, and this paper measures all three. Appendix G defines the interfaces we compare and Appendix H reports the full sweep.

Downstream algorithms. We use the two state-based algorithms reported by Park et al. (2026a). GCIVL learns a goal-conditioned value function by expectile regression and extracts a policy by advantage-weighted regression (Kostrikov et al., 2021). CRL learns a contrastive critic that is bilinear in the state and goal embeddings and extracts a policy with a behaviour-regularised deterministic objective (Eysenbach et al., 2022). The two differ in how they estimate value and in how they extract a policy, which lets us separate effects that are specific to one family from those that are not. Appendix C gives both objectives, the networks and every hyperparameter.

3 REPRESENTATION INTERVENTIONS

This section outlines our framework for isolating the policy's input interface. We first construct an exact goal-reachability representation with a controlled corruption ladder, and then design a matched set of state-side interventions to test whether the observation pathway is the true bottleneck.

3.1 GOAL-SIDE INTERVENTION

To isolate the effect of goal-representation quality, we construct an exact temporal-distance representation in the deterministic navigation environments. For each landmark a_i, we compute the shortest-path distance to every reachable maze cell using breadth-first search and represent a goal g as

$$\varphi(g) = \big[\, d(a_1, g), \ldots, d(a_K, g) \,\big] \in \mathbb{R}^K.$$

We use the same landmark dimensionality as DGR, with $K = 256$ for AntMaze and HumanoidMaze and $K = 64$ for PointMaze. The resulting representation is computed directly from the environment geometry rather than learned from the offline dataset, while the downstream learning algorithm and all other training components remain unchanged. Appendix B describes the table construction and the evaluation protocol.

Measuring representation quality. We quantify how well a representation preserves temporal-distance structure using min-plus decoding. Given a state s and representation $\varphi(g)$, we reconstruct the state-goal distance as

$$\hat{d}(s, g) = \min_i \left[d(a_i, s) + \varphi_i(g) \right]. \tag{3}$$

Representation quality is measured by the Spearman correlation between $\hat{d}(s, g)$ and the true shortest-path distance $d(s, g)$ over a fixed set of state-goal pairs. We additionally compute this correlation within distance quartiles to distinguish local from long-range structure. This provides a direct measure of the geometric information retained by the representation without fitting an auxiliary probe. Appendix D reports the correctness check on the decode and explains why a learned probe is uninformative here.

Varying representation quality. Starting from the exact representation, we progressively add noise to its distance structure using two perturbations. Gaussian perturbation modifies distances across the full range, whereas far-field perturbation modifies only distances in the furthest quartile while preserving nearby relationships:

$$\tilde{\varphi}_i(g) = \max\left(0, \varphi_i(g) + \sigma \epsilon_{i,g}\right), \qquad \epsilon_{i,g} \sim \mathcal{N}(0, 1). \tag{4}$$

The perturbation is sampled once and fixed throughout training, so each setting defines a deterministic representation rather than stochastic observation noise. Varying σ therefore produces representations with progressively different amounts and types of temporal-distance information while leaving the downstream learner unchanged.

3.2 STATE-SIDE INTERVENTION

The goal-side experiments vary what information is provided about the target. We construct a matched intervention on the agent state to determine whether the same representational structure has a different effect when applied to the current observation. For each maze cell, we define a fixed state code $c(s) \in \mathbb{R}^K$ and augment the observation as

$$\tilde{s} = \big[\, s; c(s) \,\big]. \tag{5}$$

The code uses the same dimensionality and indexing as the goal representation. We evaluate exact temporal-distance codes, corrupted variants, and a random code sampled independently for each cell. The random code preserves state identity while containing no temporal-distance structure, allowing us to distinguish the value of geometric structure from simply making the agent's position more distinguishable.

The table-based state codes require access to the maze geometry and are therefore used only as diagnostic interventions. To obtain a representation that does not depend on the map, we instead encode the agent coordinates x, y using fixed random Fourier features. With $B \in \mathbb{R}^{F\times 2}$,

$$\tilde{s} = [s; \sin(2\pi B\, xy); \cos(2\pi B\, xy)]\,. \tag{6}$$

We use $F = 128$, adding 256 fixed features to the observation. The feature matrix is sampled once and remains fixed across training, while the objective, optimiser, network widths, goal representation, and remaining hyperparameters are unchanged. This provides a simple state encoding that can be applied without access to the maze layout or transition graph.

4 Goal Representations Are Saturated

This section examines how strongly downstream performance depends on goal-representation quality. We first replace the learned representation with an exact temporal-distance representation, then progressively reduce the distance information it preserves, and finally test a representation with no meaningful distance structure. Appendix E reports the same comparisons on the remaining environments.

Table 1: **An exactly optimal goal representation does not reliably improve control.** GCIVL and CRL, concatenation interface, success rate $\times 100$, mean $\pm$ standard deviation over n seeds given in parentheses. The published column is taken from Park et al. (2026a), their Table 1 for GCIVL and Table 6 for CRL, over 8 seeds. The difference is ideal minus learned with a 95% Welch interval; bold marks intervals excluding zero.

Environment	Algo	Dual (published)	Learned (ours)	Ideal (ours)	Difference
pointmaze-med	GCIVL	76 ± 7	70.9 ± 6.1 (5)	62.7 ± 3.0 (5)	$\mathbf{-8.2\ [-15.6, -0.7]}$
pointmaze-large	GCIVL	46 ± 6	46.4 ± 6.5 (5)	46.3 ± 7.9 (5)	$-0.1\ [-10.7, +10.5]$
antmaze-med	GCIVL	75 ± 4	76.5 ± 7.4 (5)	68.4 ± 4.0 (5)	$-8.0\ [-17.2, +1.1]$
antmaze-large	GCIVL	28 ± 11	32.2 ± 8.9 (8)	30.0 ± 8.0 (8)	$-2.2\ [-11.3, +6.9]$
humanoid-med	GCIVL	29 ± 3	27.8 ± 3.6 (5)	34.2 ± 3.1 (5)	$\mathbf{+6.4\ [+1.5, +11.4]}$
pointmaze-med	CRL	33 ± 1	37.8 ± 3.3 (5)	57.4 ± 11.6 (5)	$\mathbf{+19.5\ [+5.4, +33.7]}$
pointmaze-large	CRL	39 ± 12	35.4 ± 7.9 (5)	43.0 ± 17.6 (4)	$+7.6\ [-18.8, +34.1]$
antmaze-med	CRL	93 ± 3	94.6 ± 1.6 (2)	92.5 ± 0.4 (2)	$-2.1\ [-13.8, +9.7]$
antmaze-large	CRL	87 ± 2	82.2 ± 3.4 (8)	80.8 ± 3.4 (8)	$-1.4\ [-5.1, +2.2]$

4.1 Does an Exact Goal Representation Improve Downstream Performance?

We present the comparison between the learned dual representation and the exact temporal-distance representation in Table 1, while keeping the downstream algorithm and all other training components fixed.

Of the nine evaluations, six show no statistically distinguishable difference between the two representations. Among the remaining three, the exact representation improves success by 6.4 points on humanoidmaze-medium with GCIVL and by 19.5 points on pointmaze-medium with CRL, whereas the learned representation performs 8.2 points better on pointmaze-medium with GCIVL. The differences therefore do not follow a consistent direction across environments or algorithms.

The same pattern is also reflected in the learning curves in Figure 3: replacing the learned representation with exact temporal distances does not consistently improve learning speed,

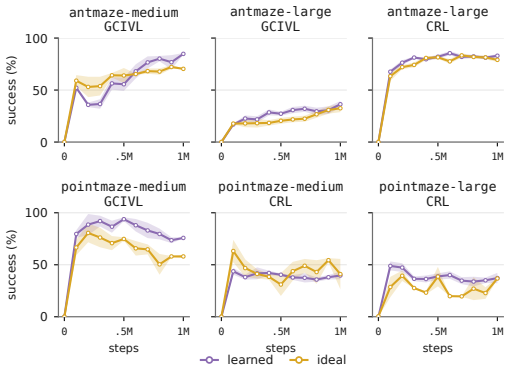


Figure 3: **Exact goal representations yield no consistent training advantage.** Evaluation curves (mean $\pm$ 1 SE over 2–8 seeds per panel; violet is learned, gold exact). Exact and learned representations exhibit indistinguishable learning rates and asymptotic success.

final performance, or the overall training trajectory. Across the five GCIVL tasks, the exact representation achieves a mean success rate of 48, compared with 51 for the learned representation.

These results extend the evaluation of exact dual goal representations beyond the tabular setting considered by Park et al. (2026a). In their Lights-Out experiment, the exact representation improves over its learned approximation, but we do not observe the same advantage consistently in the continuous-control navigation tasks studied here. More accurate temporal-distance information therefore does not, by itself, translate reliably into better downstream performance. This motivates a direct examination of whether the downstream learner makes meaningful use of the distance structure encoded by the goal representation.

4.2 How Sensitive Is Downstream Performance to Goal Representation Quality?

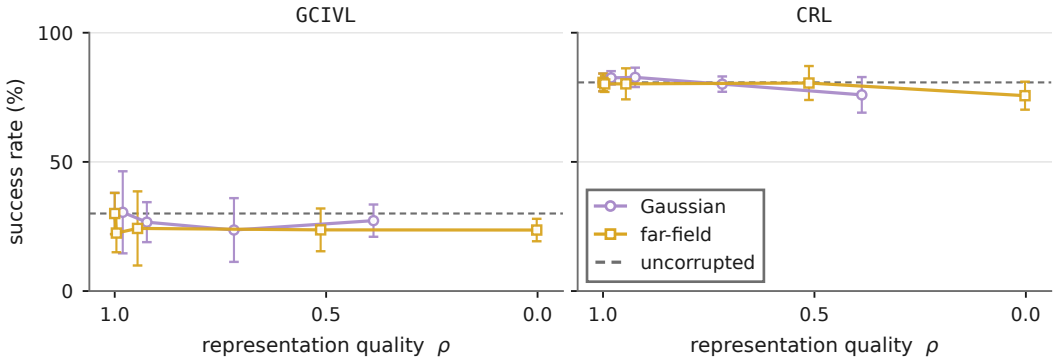


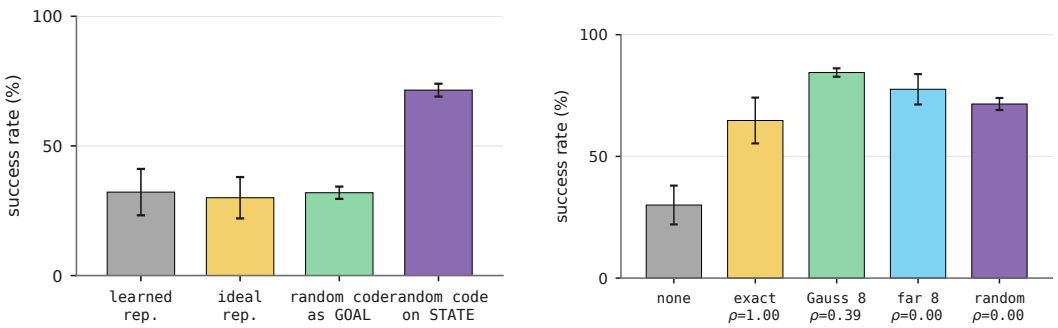
Figure 4: **Control is insensitive to goal quality.** Results on antmaze-large (mean ± 1 SD, 5–8 seeds). Across corruption ladders spanning exact geodesics to zero distance information, performance differences are statistically indistinguishable from zero.

A comparison between the learned and exact representations alone cannot determine whether representation quality matters, because the learned representation may already preserve enough of the relevant distance structure. We therefore evaluate the corruption ladder introduced in Section 3.1, progressively reducing the quality of the exact representation while keeping the downstream learner unchanged. Representation quality is measured by the Spearman correlation between the min-plus decoded distance and the true geodesic distance.

Despite spanning a wide range of representation quality, the corresponding changes in performance remain small. Under far-field corruption at $\sigma = 8$, the representation loses essentially all rank information about distant goals, yet performance decreases by only 6.4 points for GCIVL and 5.1 points for CRL. Across all sixteen settings in Figure 4, every point estimate remains within 7.6 points of the uncorrupted representation, and all confidence intervals include zero. Performance also does not vary monotonically with representation quality: under GCIVL, far-field $\sigma = 1$ produces the lowest success despite retaining the second-highest measured quality, while Gaussian $\sigma = 8$ outperforms Gaussian $\sigma = 4$. Across the corruption families, the correlation between measured representation quality and success remains between $+0.21$ and $+0.27$, with no relationships distinguishable from zero.

The same pattern holds when we distinguish between local and long-range distance information. Far-field corruption keeps the near-field correlation at 0.997 across all noise levels while reducing the far-field correlation from 1.00 to -0.41. At the strongest corruption level, nearby goals remain ordered almost perfectly while distant goals are ranked in reverse, yet the resulting performance change remains limited.

For CRL, far-field corruption changes success by only 0.2 points at $\sigma = 4$ and 5.1 points at $\sigma = 8$. These results indicate that neither local nor long-range distance structure produces a strong or consistent effect on downstream performance in this setting. We do not claim exact flatness: our stated tolerance was five points, and several confidence intervals extend beyond that range. The evidence instead supports a more limited conclusion: across a corruption ladder spanning nearly the full range



(a) (b)

Figure 5: **State-side interventions on antmaze-large (GCIVL; mean ± 1 SD, 5–8 seeds). (a)** Applying an identical fixed random $\mathcal{N}(0,1)$ code to the state more than doubles success, while doing nothing on the goal side. **(b)** The state code does not require distance geometry: corrupted and purely random codes match or outperform exact geodesics.

of measured representation quality, no performance difference is statistically distinguishable from zero, whereas the state-side interventions examined in Section 5 produce changes of roughly 30 to 50 points.

4.3 Does the Goal Representation Need to Encode Distance at All?

The ladder degrades distances but preserves the construction. The sharper question is whether the downstream agent uses distance information at all. We replace the goal representation with a table of the same shape and indexing whose entries are drawn i.i.d. from $\mathcal{N}(0,1)$. It identifies the goal cell and contains no geometry whatsoever: its rank correlation with the true distances is 0.004.

That representation scores 31.9 ± 2.4 on antmaze-large, against 30.0 ± 8.0 for the exact geodesics, $+1.9$ $[-4.9, +8.8]$, and against 32.2 ± 8.9 for DGR's learned representation, -0.2 $[-7.9, +7.4]$. A random code, an exactly optimal code and a learned code are indistinguishable (Figure 5a). Whatever these agents extract from the goal, it is little more than its identity. This also disposes of the "already good enough" reading of Section 4.1: a representation that is not good at all does just as well.

5 The State Pathway Is the Bottleneck

If the goal side is saturated, the headroom lies elsewhere. This section moves the identical intervention to the other side of the interface, identifies what the state actually needs, and proposes a method that requires no privileged information.

5.1 The Same Code Is Worth Forty Points More on the State Side

Figure 5a presents a matched pair of interventions. The construction, the dimensionality, the cell indexing and the information content are identical in the last two rows of the figure; only the argument the code is applied to differs. The gap between these last two rows is 39.6 points, produced by nothing other than which argument receives the code. The same result holds for CRL, where an exact state code raises success from 80.8 ± 3.4 to 89.9 ± 1.5, a difference of $+9.2$ $[+6.1, +12.3]$, on a task where the goal-side intervention was

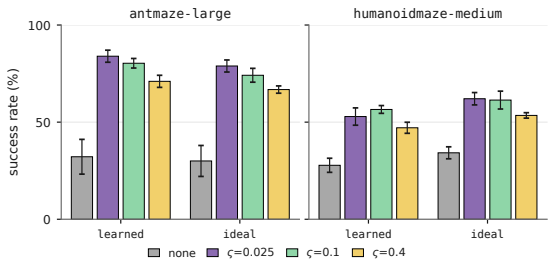


Figure 6: **Position features.** Success with and without the features of equation equation 6, for both goal representations, at frequency scales corresponding to wavelengths 8.5, 2.1 and 0.5 maze cells.

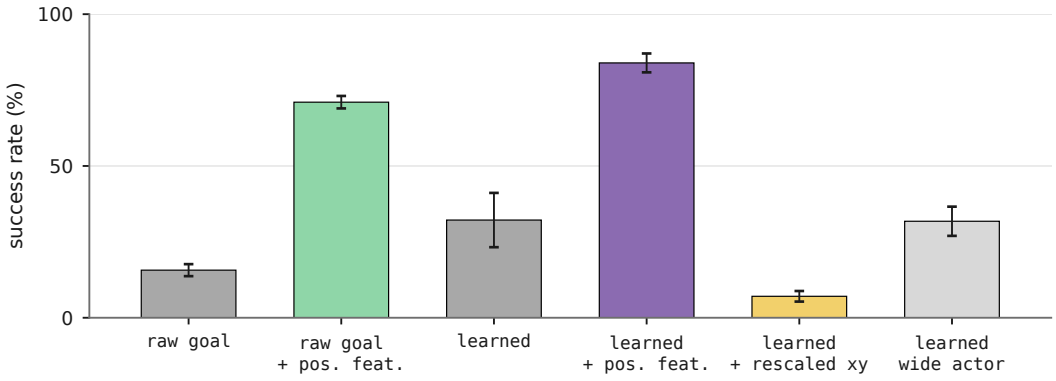


Figure 7: **Controls for the method.** antmaze-large, GCIVL. Position features work without any goal representation at all (second bar). Standardising the coordinates in place, which changes their scale without adding a basis, is severely harmful (fifth bar), and widening the actor changes nothing (sixth bar).

worth -1.4 $[-5.1, +2.2]$. Appendix F tabulates this and the remaining state-side comparisons.

5.2 What Must the State Code Contain?

Having found that the state side matters, we ask what it actually needs to contain. Figure 5b explores this by running the state code through the same corruption ladder used for the goal, alongside the random table. Every code helps, and the exact geodesics are the weakest of them. Corrupting the table improves it, by 19.8 points for Gaussian $\sigma = 8$ and 12.9 points for far-field $\sigma = 8$, and a table with no geometry whatsoever is worth 6.8 points more than the exact one, $+6.8$ $[-4.8, +18.3]$. The effect is therefore positional encoding: what the network gains is a high-dimensional, distinguishable representation of where it currently is, and the exact geodesic table is a comparatively poor one because it varies smoothly and nearly linearly across neighbouring cells.

5.3 Position Features

While discrete state codes require access to the maze layout, the position features in equation 6 require only the agent's raw coordinates. Figure 6 evaluates this intervention on the two hardest navigation tasks across both goal representations and three frequency scales.

On antmaze-large, the method raises GCIVL success from 32.2 to 83.9 when paired with DGR's learned representation. By comparison, published goal-representation methods span 9 to 28 on this task (Park et al., 2026a, Table 1). On humanoidmaze-medium, it roughly doubles success under both representations. Across all settings, lower frequencies perform best, while the highest frequency performs worst (Figure 10).

Ablations in Figure 7 isolate the source of this gain. Crucially, position features succeed without any goal representation at all, lifting a raw-goal baseline from 15.7 to 71.0. DGR's learned representation provides an additional $+12.9$ $[+9.0, +16.9]$ on top of them, indicating that while goal representations are not useless, their contribution is second-order. Standardising the coordinates in place, which changes their scale without adding a basis, does not reproduce the gain and is severely harmful.

5.4 Ruling Out Confounding Mechanisms

To understand why state-side intervention drives performance, we systematically evaluated and ruled out several candidate mechanisms:

- **Model capacity:** Widening the actor to 1024 hidden units changes nothing (-0.4 $[-8.8, +8.0]$), and the best-performing interface uses fewer parameters (635k) than standard concatenation (676k).
- **Input scale:** Standardising coordinates drops success from 32.2 to 7.1, confirming that position requires high-frequency distinguishability rather than standard conditioning.
- **Goal input width:** Compressing $\varphi(g)$ to match the attention context width produces no discernible change (-2.8 $[-13.5, +7.8]$ with the ideal representation; -0.1 $[-6.5, +6.3]$ with raw goals).
- **Internal shortest-path routing:** Cross-attention over landmark tokens does not implement min-plus decoding equation 3. Attention peaks align with a state-independent goal-proximity heuristic rather than true min-plus argmins.

Interface corroboration and scope. Combining state and goal via cross-attention improves antmaze-large performance by $+23.2$ $[+15.6, +30.9]$ because queries are formed directly from state observations. Crucially, this gain is redundant when an explicit state code is already present ($+8.9$ $[-3.4, +21.2]$). These benefits remain specific to complex navigation: they do not transfer to manipulation tasks or low-dimensional environments like pointmaze, where coordinates already dominate the state vector. We provide complete attention diagnostics, parameter sweeps, and scope evaluations in Appendix I.

6 Conclusion

We investigated whether goal-representation quality bounds performance in offline goal-conditioned navigation. Across our benchmarks, replacing learned goal embeddings with exact shortest-path representations, degrading their distance structure, or substituting random goal codes does not substantially change downstream task success. In contrast, applying the same interventions to the agent's observation more than doubles success on antmaze-large, and a map-free Fourier positional encoding raises GCIVL from 32 to 84 while roughly doubling performance on humanoidmaze-medium. These findings indicate that the state pathway, rather than the goal representation, is the primary performance bottleneck.

7 Limitations

Our findings carry several limitations: while we rule out capacity, input scale, and internal shortest-path routing, the precise mechanism explaining why spatial distinguishability aids policy learning remains an open question. Furthermore, our evaluation is restricted to state-based navigation; the benefits do not transfer to manipulation tasks or low-dimensional environments like pointmaze, where coordinates already dominate observations (Appendix J). Finally, we do not claim state-of-the-art results over hierarchical methods, and position features remain untested in pixel-based domains.

AI Use Statement

Generative AI tools were used to assist with language editing, rewriting, and improving the clarity and presentation of portions of the manuscript. They were also used to help organise and refine descriptions of the experimental methodology and results. All scientific claims, experimental results, interpretations, and conclusions were determined and verified by the authors. AI-assisted text was reviewed and revised by the authors to ensure consistency with the underlying experiments and reported results. No generative AI system was treated as an authoritative source of scientific evidence or used to evaluate the validity of the experimental findings. The authors take full responsibility for the final content of the paper produced with the assistance of generative AI.

Ethics Statement

This work studies representation learning and offline goal-conditioned reinforcement learning using existing simulated benchmark environments. The experiments do not involve human participants,

personally identifiable information, or sensitive real-world data. We do not identify any direct ethical risks specific to the methods or experiments presented in this work beyond the broader considerations associated with reinforcement learning and machine learning research.

Reproducibility Statement

We provide the methodological and experimental details required to reproduce our results throughout the main paper and appendices. Section 3 describes the construction of the exact temporal-distance representations, the representation-quality metric, the corruption procedures, the state-side interventions, and the positional features. The experimental protocol specifies the environments, downstream algorithms, evaluation procedure, number of seeds, and statistical comparisons used throughout the study. Additional implementation details, ablations, and results are provided in the appendices. We will also provide the code and experiment configurations necessary to reproduce the reported results in the camera-ready version.

References

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G. Bellemare. Deep reinforcement learning at the edge of the statistical precipice, 2022. URL https://arxiv.org/abs/2108.13264.

Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep variational information bottleneck, 2019. URL https://arxiv.org/abs/1612.00410.

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 5055–5065, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Pablo Samuel Castro, Tyler Kastner, Prakash Panangaden, and Mark Rowland. Mico: Improved representations via sampling-based state similarity for markov decision processes. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 30113–30126. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/fd06b8ea02fe5b1c2496fe1700e9d16c-Paper.pdf.

Elliot Chane-Sane, Cordelia Schmid, and Ivan Laptev. Goal-conditioned reinforcement learning with imagined subgoals, 2021. URL https://arxiv.org/abs/2107.00541.

Cédric Colas, Olivier Sigaud, and Pierre-Yves Oudeyer. How many random seeds? statistical power analysis in deep reinforcement learning experiments, 2018. URL https://arxiv.org/abs/1806.08295.

Ben Eysenbach, Russ R Salakhutdinov, and Sergey Levine. Search on the replay buffer: Bridging planning and reinforcement learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/5c48ff18e0a47baaf81d8b8ea51eec92-Paper.pdf.

Benjamin Eysenbach, Tianjun Zhang, Sergey Levine, and Ruslan Salakhutdinov. Contrastive learning as goal-conditioned reinforcement learning. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Dibya Ghosh, Abhishek Gupta, Ashwin Reddy, Justin Fu, Coline Devin, Benjamin Eysenbach, and Sergey Levine. Learning to reach goals via iterated supervised learning, 2020. URL https://arxiv.org/abs/1912.06088.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 21271–21284. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/f3ada80d5c4ee70142b17b8192b2958e-Paper.pdf.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters, 2019. URL https://arxiv.org/abs/1709.06560.

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

Siddhant M. Jayakumar, Wojciech M. Czarnecki, Jacob Menick, Jonathan Schwarz, Jack Rae, Simon Osindero, Yee Whye Teh, Tim Harley, and Razvan Pascanu. Multiplicative Interactions and Where to Find Them. In *International Conference on Learning Representations*, 2020. URL https://mlanthology.org/iclr/2020/jayakumar2020iclr-multiplicative/.

Leslie Pack Kaelbling. Learning to Achieve Goals. In *International Joint Conference on Artificial Intelligence*, pp. 1094–1099, 1993. URL https://mlanthology.org/ijcai/1993/kaelbling1993ijcai-learning/.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning, 2021. URL https://arxiv.org/abs/2110.06169.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems, 2020. URL https://arxiv.org/abs/2005.01643.

Yecheng Jason Ma, Shagun Sodhani, Dinesh Jayaraman, Osbert Bastani, Vikash Kumar, and Amy Zhang. Vip: Towards universal visual reward and representation via value-implicit pre-training, 2023. URL https://arxiv.org/abs/2210.00030.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: representing scenes as neural radiance fields for view synthesis. *Commun. ACM*, 65(1):99–106, December 2021. ISSN 0001-0782. doi: 10.1145/3503250. URL https://doi.org/10.1145/3503250.

Vivek Myers, Catherine Ji, and Benjamin Eysenbach. Horizon generalization in reinforcement learning. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 76302–76326, 2025a. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/bddd4e769412635c62edd8e74a413e28-Paper-Conference.pdf.

Vivek Myers, Bill Zheng, Anca Dragan, Kuan Fang, and Sergey Levine. Temporal representation alignment: Successor features enable emergent compositionality in robot instruction following. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 149934–149961. Curran Associates, Inc., 2025b. doi: 10.52202/085713-5013. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/dc8fe7925e37b2fa764d0ec3294c9133-Paper-Conference.pdf.

Vivek Myers, Bill Chunyuan Zheng, Benjamin Eysenbach, and Sergey Levine. Offline goal-conditioned reinforcement learning with quasimetric representations, 2025c. URL https://arxiv.org/abs/2509.20478.

Ofir Nachum, Shixiang Gu, Honglak Lee, and Sergey Levine. Data-efficient hierarchical reinforcement learning. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pp. 3307–3317, Red Hook, NY, USA, 2018. Curran Associates Inc.

Seohong Park, Dibya Ghosh, Benjamin Eysenbach, and Sergey Levine. Hiql: Offline goal-conditioned rl with latent states as actions, 2024a. URL https://arxiv.org/abs/2307.11949.

Seohong Park, Tobias Kreiman, and Sergey Levine. Foundation policies with hilbert representations, 2024b. URL https://arxiv.org/abs/2402.15567.

Seohong Park, Kevin Frans, Benjamin Eysenbach, and Sergey Levine. Ogbench: Benchmarking offline goal-conditioned rl. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 94937–94982, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/ecd92623ac899357312aaa8915853699-Paper-Conference.pdf.

Seohong Park, Deepinder Mann, and Sergey Levine. Dual goal representations. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 153307–153324, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/f8cd7eb180deb5c6ec876de2b442fa72-Paper-Conference.pdf.

Seohong Park, Aditya Oberai, Pranav Atreya, and Sergey Levine. Transitive rl: Value learning via divide and conquer, 2026b. URL https://arxiv.org/abs/2510.22512.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. Film: visual reasoning with a general conditioning layer. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018. ISBN 978-1-57735-800-8.

Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred A. Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks, 2019. URL https://arxiv.org/abs/1806.08734.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Proceedings of the 21st International Conference on Neural Information Processing Systems*, NIPS'07, pp. 1177–1184, Red Hook, NY, USA, 2007. Curran Associates Inc. ISBN 9781605603520.

Nikolay Savinov, Alexey Dosovitskiy, and Vladlen Koltun. Semi-parametric topological memory for navigation. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=SygwwGbRW.

Tom Schaul, Dan Horgan, Karol Gregor, and David Silver. Universal value function approximators. In *Proceedings of the 32nd International Conference on Machine Learning - Volume 37*, ICML'15, pp. 1312–1320. JMLR.org, 2015.

Max Schwarzer, Ankesh Anand, Rishab Goel, R Devon Hjelm, Aaron Courville, and Philip Bachman. Data-efficient reinforcement learning with self-predictive representations, 2021. URL https://arxiv.org/abs/2007.05929.

Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, and Sergey Levine. Time-contrastive networks: Self-supervised learning from video, 2018. URL https://arxiv.org/abs/1704.06888.

Vincent Sitzmann, Julien N. P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions, 2020. URL https://arxiv.org/abs/2006.09661.

Aravind Srinivas, Michael Laskin, and Pieter Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning, 2020. URL https://arxiv.org/abs/2004.04136.

Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. URL https://arxiv.org/abs/1706.03762.

Tongzhou Wang and Phillip Isola. On the learning and learnability of quasimetrics, 2022. URL https://arxiv.org/abs/2206.15478.

Tongzhou Wang, Antonio Torralba, Phillip Isola, and Amy Zhang. Optimal goal-reaching reinforcement learning via quasimetric learning, 2023. URL https://arxiv.org/abs/2304.01203.

A Related Work

Offline goal-conditioned RL. Goal-conditioned policies trained to reach arbitrary states date back to Kaelbling (1993) and were scaled with universal value approximators (Schaul et al., 2015) and hindsight relabelling (Andrychowicz et al., 2017; Ghosh et al., 2020). The offline setting inherits the distribution-shift problems of offline RL (Levine et al., 2020; Kumar et al., 2020) and is typically addressed with in-sample value learning (Kostrikov et al., 2021). We use the state-based navigation tasks and the two downstream algorithms of OGBench (Park et al., 2025), which also ships oracle goal-representation variants of these tasks; those variants are reported by Park et al. (2026b) but were not benchmarked when introduced.

Goal representations from temporal structure. The premise we test is that a goal is best described by its "reachability" relations. Contrastive objectives learn such structure from future occupancy (Eysenbach et al., 2022; Sermanet et al., 2018), value-implicit pre-training learns it from value functions (Ma et al., 2023), and quasimetric and Hilbert embeddings impose the metric structure architecturally (Wang et al., 2023; Wang & Isola, 2022; Park et al., 2024b; Myers et al., 2025c;b). Dual goal representations extend this idea, describing a goal by its temporal distances to all other states (Park et al., 2026a). The appeal of distance-structured goals is that such distances compose over the horizon (Myers et al., 2025a). Our contribution is a measurement of what this family buys: we construct the exact representation these methods approximate and find that downstream control is insensitive to it. Other goal or state representation objectives, including information bottlenecks (Alemi et al., 2019), self-predictive and contrastive auxiliary losses (Grill et al., 2020; Schwarzer et al., 2021; Srinivas et al., 2020), and behavioural metrics (Castro et al., 2021), are the comparison columns we quote from Park et al. (2026a).

What else has been identified as the bottleneck. A parallel line of work locates the difficulty in the horizon rather than the representation, and reduces it with hierarchy or n-step returns (Park et al., 2024a; Nachum et al., 2018), or with explicit search over the replay buffer (Eysenbach et al., 2019; Savinov et al., 2018; Chane-Sane et al., 2021). These methods remain stronger than ours in absolute terms on the hardest mazes (Park et al., 2025). Our finding is complementary: where that literature shortens the decision problem, we change how the agent's own position enters the network.

Input encodings and conditioning. Random features (Rahimi & Recht, 2007) and Fourier features (Tancik et al., 2020) are standard tools for making coordinate inputs learnable, and are central to implicit neural representations (Mildenhall et al., 2021; Sitzmann et al., 2020); the usual justification is spectral bias (Rahaman et al., 2019), which our frequency sweep does not support. How two inputs are combined is itself a design axis, studied as feature-wise modulation (Perez et al., 2018), attention (Vaswani et al., 2023), and multiplicative interactions (Jayakumar et al., 2020), but it is fixed to concatenation throughout the goal-representation literature. Finally, our emphasis on intervals, seed counts and pre-registered thresholds follows recommended practice for empirical RL (Henderson et al., 2019; Agarwal et al., 2022; Colas et al., 2018), and our use of a decoder rather than a trained probe follows the control-task critique of probing (Hewitt & Liang, 2019).

B Experimental Setup and Protocol

Tasks and datasets. We use the state-based navigation tasks of OGBench (Park et al., 2025): pointmaze-medium, pointmaze-large, antmaze-medium, antmaze-large and humanoidmaze-medium, with the standard `navigate` datasets. In these tasks temporal distance is a graph geodesic, which is what makes the ideal representation computable. Observations are 4, 4, 29, 29 and 69 dimensional, and two dimensions hold the agent's position in every case.

Training and evaluation. Every run trains for one million gradient steps. We evaluate on the five held-out goals of the benchmark with fifty episodes each, every 100k steps, and report the mean success rate over the final three evaluations at 800k, 900k and one million steps. We never report the best evaluation over training. A single run takes between two and twelve hours on one RTX 4090 or RTX 3090.

Reporting. Arms are compared with Welch's unequal-variance t-test and 95% confidence intervals, and every table and figure states its seed count. Because several of our central claims are null

results, we report intervals rather than p-values alone, following recommended practice for empirical reinforcement learning (Henderson et al., 2019; Agarwal et al., 2022; Colas et al., 2018).

Constructing the exact representation. We sample K landmark states uniformly from the dataset, map each one to the maze cell that contains it, and run a breadth-first search from every landmark to every cell. The representation has to be a function of the raw observation rather than a column of the dataset, because at evaluation time the environment supplies goals that have no dataset index. We therefore store the affine map from coordinates to cell indices next to the table, so the same lookup serves dataset goals and evaluation goals. Cells occupied by walls are filled by nearest neighbour, so the lookup is always defined. On antmaze-large this gives a table over 736 cells drawn from 186 distinct landmark cells, and every state in the dataset maps to a free cell. The scale of the resulting vector is a free choice that the learned representation makes implicitly. We make it explicit and standardise each coordinate, and report the alternatives in Appendix E.

C Architecture and Implementation Details

This section describes the networks, the two downstream algorithms and the inputs we intervene on. Anything not stated here follows the reference implementation of Park et al. (2026a), and all arms within a comparison share it.

C.1 Networks

Every network is a multilayer perceptron with three hidden layers of 512 units, GELU activations and layer normalisation. The value function is an ensemble of two heads and is trained against a target copy that tracks the online parameters at rate $\tau = 0.005$. The actor outputs a Gaussian with a state-independent standard deviation. The learned dual goal representation comes from a separate network of the same shape, trained on the offline data with a bilinear value parameterisation, and it enters the downstream learner through a stop-gradient. Goals are supplied by hindsight relabelling with the ratios of the benchmark. Training uses Adam with a learning rate of 3×10^{-4} and a batch size of 1024. Table 2 lists the values we used.

Table 2: **Hyperparameters.** Shared values follow the benchmark. Algorithm-specific and method-specific values are grouped below them.

Hyperparameter	Value
Optimiser	Adam
Learning rate	3×10^{-4}
Batch size	1024
Gradient steps	10^6
Hidden dimensions (value, actor, representation)	$(512, 512, 512)$
Nonlinearity	GELU
Layer normalisation	yes
Target update rate τ	0.005
Discount γ	0.99, and 0.995 on humanoidmaze-medium
Representation width K	256 on antmaze and humanoidmaze, 64 on pointmaze
Representation parameterisation	bilinear
Representation expectile	0.9
GCIVL value expectile	0.9
GCIVL advantage temperature α	10.0
GCIVL value goal ratios (current, trajectory, random)	$(0.2, 0.5, 0.3)$
CRL critic latent dimension	512
CRL behaviour-cloning coefficient α	0.1
CRL value goal ratios (current, trajectory, random)	$(0.0, 1.0, 0.0)$
Actor goal ratios (current, trajectory, random)	$(0.0, 1.0, 0.0)$
Position features F	128, giving 256 appended dimensions
Frequency scale ς (cycles per coordinate unit)	0.025, 0.1, 0.4

C.2 Downstream Algorithms

GCIVL. GCIVL fits a goal-conditioned value function by expectile regression on the benchmark reward and extracts a policy by advantage-weighted regression (Kostrikov et al., 2021). With target value $\bar{V}$ and expectile κ,

$$\begin{aligned} \mathcal{L}_V &= \mathbb{E}_{s,s',g\sim\mathcal{D}}\Big[\, \ell_\kappa\big(r(s,g) + \gamma \bar{V}(s',\varphi(g)) - V(s,\varphi(g))\big)\,\Big], \\ \mathcal{L}_\pi &= -\mathbb{E}_{s,a,g\sim\mathcal{D}}\Big[\exp\big(\alpha\, A(s,a,g)\big)\, \log \pi\big(a \mid s,\varphi(g)\big)\,\Big], \end{aligned} \tag{7}$$

where $\ell_\kappa(u) = |\kappa - \mathbf{1}[u<0]|\, u^2$ is the expectile loss, A is the advantage implied by the value function, and the exponentiated advantage is clipped at 100. We keep the benchmark implementation, in which the value function is an ensemble of two heads and the expectile weight is taken from the target advantage.

CRL. CRL fits a critic that is bilinear in a state-action embedding and a goal embedding, and trains it as a binary classifier that separates goals reached later in the same trajectory from goals taken from elsewhere in the batch (Eysenbach et al., 2022). With $f(s,a,g) = u(s,a)^\top v(\varphi(g))/\sqrt{m}$ over a batch of size B,

$$\begin{aligned} \mathcal{L}_f &= -\frac{1}{B^2}\sum_{i,j}\Big[\, \mathbf{1}[i=j]\log\sigma\big(f_{ij}\big) + \mathbf{1}[i\neq j]\log\big(1-\sigma(f_{ij})\big)\,\Big], \\ \mathcal{L}_\pi &= -\mathbb{E}\Big[\, f\big(s,\mu(s,\varphi(g)),g\big)\,\Big]/\mathbb{E}\big[|f|\big] \;-\; \alpha\,\mathbb{E}\Big[\log\pi\big(a \mid s,\varphi(g)\big)\,\Big], \end{aligned} \tag{8}$$

so the policy maximises a scale-normalised critic value with a behaviour-cloning term of weight α. The two algorithms differ in how they estimate value and in how they extract a policy, which lets us separate effects that are specific to one family from those that are not.

C.3 The Inputs We Intervene On

All five inputs below enter the same networks through the same concatenation interface, unless an experiment states otherwise. The objective, the optimiser, the network widths and the training length never change.

Learned representation. The dual goal representation of Park et al. (2026a), produced by the bilinear representation network described above. This is the baseline.

Ideal representation. Equation equation 1 evaluated with exact breadth-first distances, standardised per coordinate. Nothing about it is learned.

Corrupted representations. The ideal representation with the noise of equation equation 4 added at a fixed seed, either to all entries or to the furthest quartile only. Each setting is a deterministic representation, not observation noise.

State codes. A fixed vector per maze cell appended to the observation as in equation equation 5. The code is either the exact distance row, a corrupted row, or a random $\mathcal{N}(0,1)$ draw. These require the maze layout and are diagnostics, not a method.

Position features. Equation equation 6. We draw $B \in \mathbb{R}^{F\times 2}$ once from $\mathcal{N}(0,\varsigma^2)$ with $F = 128$, keep it fixed across seeds, and append the sine and cosine of the projected coordinates to the observation. This is the standard random Fourier construction (Rahimi & Recht, 2007; Tancik et al., 2020) applied to the two observation dimensions that hold the agent's position. It needs no map and no change to the objective, and the only thing it changes about the agent is the width of the observation.

D Measuring Representation Quality

When every cell is a landmark, the min-plus decode of equation equation 3 reproduces the true geodesic to 9.5×10^{-7}, which is one unit in the last place of single precision. We use this as a joint correctness check on the table, the indexing and the decode.

Why not a probe. Regressing temporal distance from a representation is the natural alternative, and it works in other domains. It does not work here. On antmaze-large a multilayer probe scores between 0.997 and 0.999 on a raw two-dimensional goal, on a frozen random projection and on a learned representation alike, and a bilinear probe ranks a random 256-dimensional projection above the raw goal. The reason is structural. In a maze the goal state determines its own position, so any representation that preserves goal identity contains complete distance information, and the question of how much distance is decodable is saturated by construction. This is the control-task problem in probing (Hewitt & Liang, 2019): a probe strong enough to recover the quantity can recover it from a representation that does not encode it. Min-plus decoding avoids this because it reads the representation directly and fits nothing.

E Additional Goal-Side Results

E.1 The Full Navigation Suite

Evaluating downstream performance ceilings across all five environments and both algorithms underscores that representation exactness does not drive control (Figure 8; Table 1).

Across these settings, the exact goal representation ties the learned baseline in most cases. Where differences do appear, they switch signs between algorithms and remain small relative to run-to-run seed variance. Across all five environments, providing exact distance information fails to unlock performance that the learned representation could not already achieve.

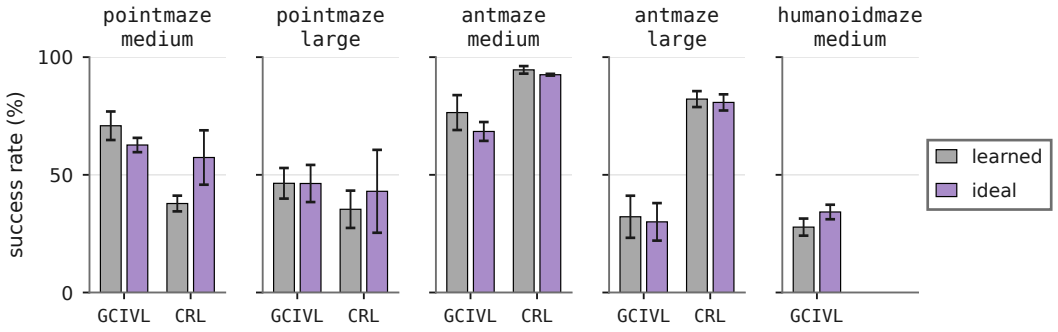


Figure 8: **Exact and learned goal representations across the full navigation suite.** Mean ± 1 SD over 5 to 8 seeds. CRL was not run on humanoidmaze-medium.

E.2 Corruption Beyond antmaze-large

Table 3 repeats the far-field ladder in three more settings. On antmaze-medium the arms stay together under both algorithms, and on pointmaze-large under GCIVL the corrupted arms are if anything slightly above the exact one. None of the three shows a monotone decline with corruption strength.

The exception is pointmaze-large under CRL, where the uncorrupted arm scores 43.0 ± 17.6 and the corrupted arms range from 14.7 to 31.5. This cell has the widest seed variance anywhere in our data, and each corrupted setting has two seeds. We report it for completeness and do not read a dependence on representation quality out of it.

Table 3: **Far-field corruption beyond antmaze-large.** Success rate $\times 100$, mean ± 1 SD over n seeds. Larger σ removes more long-range distance structure.

Environment	Algo	Uncorrupted	$\sigma = 1$	$\sigma = 2$	$\sigma = 4$	$\sigma = 8$
antmaze-medium	GCIVL	68.4 ± 4.0 (5)	64.9 ± 3.4 (2)	69.2 ± 2.5 (2)	71.5 ± 5.6 (2)	70.7 ± 0.6 (2)
antmaze-medium	CRL	92.5 ± 0.4 (2)	92.1 ± 2.0 (2)	93.1 ± 0.4 (2)	91.1 ± 1.6 (2)	90.5 ± 5.2 (2)
pointmaze-large	GCIVL	46.3 ± 7.9 (5)	50.7 ± 4.1 (2)	53.4 ± 7.3 (2)	55.0 ± 4.8 (2)	49.3 ± 0.7 (2)
pointmaze-large	CRL	43.0 ± 17.6 (4)	14.9 ± 2.7 (2)	14.7 ± 7.2 (2)	29.5 ± 12.1 (2)	31.5 ± 4.9 (2)

E.3 What the Two Corruptions Do to the Geometry

The two corruption families have differing effects on the goal and state representation. This is portrayed in Figure 9. We decode the distance with equation equation 3 and correlate it against the true geodesic within each quartile of true distance. Gaussian corruption degrades every range at once. Far-field corruption leaves the nearest quartile exact at every level while driving the furthest quartile negative, so distant goals end up ranked in reverse. The two therefore damage the representation in different ways, and they arrive at the same place downstream (Figure 4).

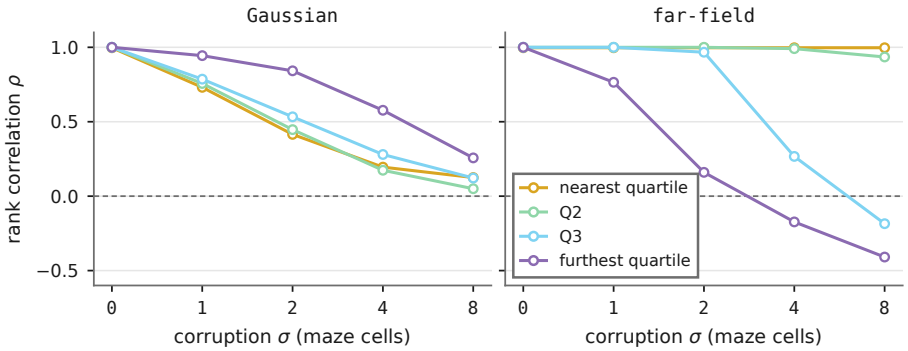


Figure 9: **Effect of the two corruption procedures on temporal-distance structure.** Spearman correlation between the decoded and true distance, within quartiles of true distance. Gaussian corruption degrades all ranges; far-field corruption keeps the nearest quartile exact and reverses the furthest.

E.4 Normalisation of the Exact Representation

The scale of the exact distance vector is a free choice. Table 4 compares the scheme used everywhere else in the paper against unit length and against raw step counts. Length normalisation is worth more to both algorithms than the difference between the exact and the learned representation. This is the same theme as the state-side result: how the vector is presented to the network matters more than the geometry inside it.

Table 4: **Normalisation of the exact table on antmaze-large.** Differences are against per-coordinate standardisation, with a 95% Welch interval.

Algo	Standardised	Unit ℓ_2	Difference	Raw counts	Difference
GCIVL	30.0 ± 8.0 (8)	41.8 ± 8.1 (3)	$+11.8$ $[-4.2, +27.8]$	28.7 ± 8.5 (3)	-1.3 $[-18.2, +15.5]$
CRL	80.8 ± 3.4 (8)	86.2 ± 2.9 (3)	$+5.5$ $[-0.1, +11.0]$	74.9 ± 6.1 (3)	-5.9 $[-19.3, +7.6]$

E.5 Number of Landmarks

To test whether the width of the representation matters, we halve and quadruple the landmark count. This leaves control unchanged in every case we ran (Table 5). This also rules out landmark count as the reason the state-side and interface effects appear on antmaze but not on pointmaze, since pointmaze at $K = 256$ behaves like pointmaze at $K = 64$.

Table 5: **Landmark count with the ideal representation.** Concatenation interface. Differences are the larger count minus the smaller, with a 95% Welch interval.

Environment	Algo	$K = 64$	$K = 256$	Difference
antmaze-large	GCIVL	28.5 ± 6.9 (5)	30.0 ± 8.0 (8)	$+1.5$ $[-7.9, +10.9]$
antmaze-large	CRL	82.6 ± 4.1 (3)	80.8 ± 3.4 (8)	-1.8 $[-10.1, +6.4]$
pointmaze-medium	GCIVL	62.7 ± 3.0 (5)	66.7 ± 6.0 (6)	$+4.0$ $[-2.6, +10.6]$
pointmaze-large	GCIVL	46.3 ± 7.9 (5)	49.0 ± 6.7 (7)	$+2.6$ $[-7.4, +12.7]$

F Additional State-Side Results

F.1 Results the Main Text Refers To

Several additional experiments clarify how and where state-side interventions help, as summarised in Table 6.

Three results stand out. First, state codes generalise beyond GCIVL: applying them to CRL improves performance by $+9.2$ points on the same task where goal-side intervention achieved nothing. Second, reading the ideal goal representation with cross-attention across both the actor and value networks adds $+33.2$ points, but only $+3.9$ with the learned representation. Downstream control can exploit higher-quality goal information only when given an interface capable of extracting it. Third, adding cross-attention to an agent that already has an explicit state code yields only $+8.9$ points (not distinguishable from zero), compared to $+23.2$ points when no state code is present. The two interventions provide redundant benefits.

Table 6: **State-side and interface results referenced in the main text.** All on antmaze-large. Differences are against the baseline column with a 95% Welch interval; bold marks intervals excluding zero.

Configuration	Baseline	With intervention	Difference
CRL, exact state code	80.8 ± 3.4 (8)	89.9 ± 1.5 (5)	$\mathbf{+9.2}$ $\mathbf{[+6.1, +12.3]}$
GCIVL, state code + cross-attention	64.7 ± 9.4 (5)	73.7 ± 7.0 (5)	$+8.9$ $[-3.4, +21.2]$
GCIVL, ideal + x-attn on value and actor	30.0 ± 8.0 (8)	63.2 ± 5.0 (5)	$\mathbf{+33.2}$ $\mathbf{[+25.2, +41.1]}$
GCIVL, learned + x-attn on value and actor	32.2 ± 8.9 (8)	36.1 ± 3.3 (5)	$+3.9$ $[-3.9, +11.7]$

F.2 Which Wavelengths Work

Performance varies systematically with feature wavelength in all four settings (Figure 10).

Longer wavelengths consistently perform better, peaking at roughly eight maze cells. Short wavelengths act like random codes; they separate states but drop all spatial proximity. In contrast, long wavelengths keep nearby states similar while separating distant ones, which helps the policy most. Because our sweep stopped at eight cells, where this trend eventually peaks remains open.

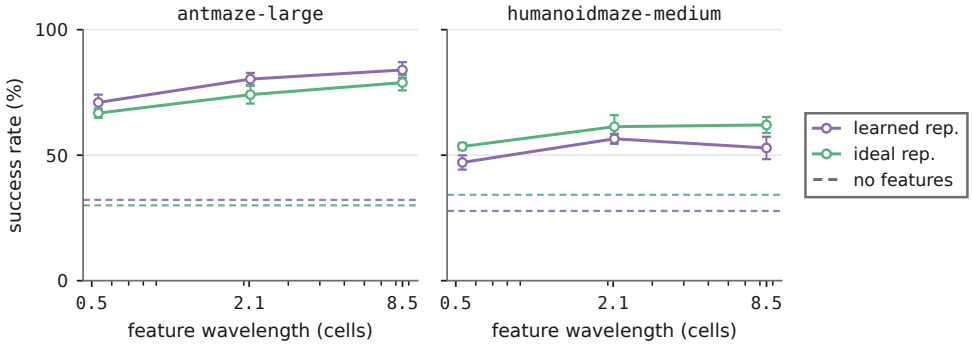


Figure 10: **Success against feature wavelength.** The runs of Figure 6, plotted against wavelength in maze cells. Dashed lines in the matching colour mark the same agent without position features.

G Interface Definitions

Concatenation is the baseline used by all prior work, equation equation 2. FiLM computes modulation parameters from the goal and applies them to a state-only trunk (Perez et al., 2018),

$$h \leftarrow h \odot \big(1 + \gamma(\varphi)\big) + \beta(\varphi). \tag{9}$$

Cross-attention treats the K coordinates of φ as tokens. Token i is a learned landmark embedding added to an embedding of the scalar $\varphi_i(g)$, a query is formed from the state, and the attended context is concatenated with the state before the trunk (Vaswani et al., 2023),

$$\begin{aligned} c &= \mathrm{Attn}\big(q(s),\ \{\, e_i + W\varphi_i(g)\,\}_{i=1}^{K}\big), \\ I_{\mathrm{xattn}}(s,\varphi) &= \mathrm{MLP}\big([\,s\,;\,c\,]\big). \end{aligned} \tag{10}$$

We use four heads and a width of 64 for the tokens and the context. The bottleneck control compresses φ to the same width with a plain network and then concatenates, so any gain that cross-attention gets merely by narrowing the goal input should appear there as well.

Two properties keep the comparison fair. First, both new interfaces start as a state-only function: the FiLM modulation begins at the identity because γ and β are zero-initialised, and the attention output projection is zero-initialised. Any gain they obtain therefore comes from learning to use the goal and not from a different function at initialisation. Concatenation and the bottleneck pass the goal through a randomly initialised first layer, as in prior work. Second, the interfaces are close in size, at 676k parameters for concatenation, 635k for cross-attention, 660k for the bottleneck and 1.47M for FiLM, so the best interface is not the largest.

H Full Interface Sweep

How the policy ingests goal representations often matters more than the representation itself (Figure 11). Across fifteen controlled comparisons holding representations, algorithms, and hyperparameters fixed, replacing standard concatenation with cross-attention shifts performance from -8.4 to $+26.9$ points, with six runs cleanly separated from zero (five favoring cross-attention). Rather than advocating cross-attention as a standalone technique, this comparison demonstrates that standard architectural defaults, often fixed by convention, introduce larger performance shifts than optimising goal representations.

I Mechanistic Controls

This section rules out the explanations of the state-side gain that do not involve spatial distinguishability.

I.1 Capacity, Scale and Input Width

Capacity. Doubling the actor width from 512 to 1024 units per hidden layer under default GCIVL changes nothing, at -0.4 $[-8.8, +8.0]$. Across the interfaces, parameter count does not track performance either: the best variant has 635k parameters against 676k for concatenation, and the largest at 1.47M is the worst.

Coordinate scale. If the gain came from numerical conditioning, then fixing the scale of the coordinates should be enough. It is not. Standardising (x, y) to zero mean and unit variance, which is the smallest intervention that aligns position scale with the proprioceptive dimensions, drops success from 32.2 to 7.1. The features help by separating positions, not by rescaling them.

Input width. A high-dimensional goal vector might help simply by being wide. Passing $\varphi(g)$ through a linear bottleneck matched to the attention context width changes nothing, both with the ideal representation at -2.8 $[-13.5, +7.8]$ and with raw goal observations at -0.1 $[-6.5, +6.3]$.

I.2 The Policy Does Not Route Through Landmarks

Cross-attention over landmark tokens is structurally able to carry out the min-plus decode of equation equation 3, so we checked whether it does. We extracted the attention distribution over 256 landmark tokens for 4096 evaluation state-goal pairs from two independently trained seeds.

Attention puts $1.59\times$ and $1.43\times$ chance mass on the landmark that the true min-plus decode selects, which looks supportive until compared with a null. A state-independent baseline that always picks the landmark nearest the goal receives more mass, at $2.52\times$ and $1.85\times$ chance. The min-plus path

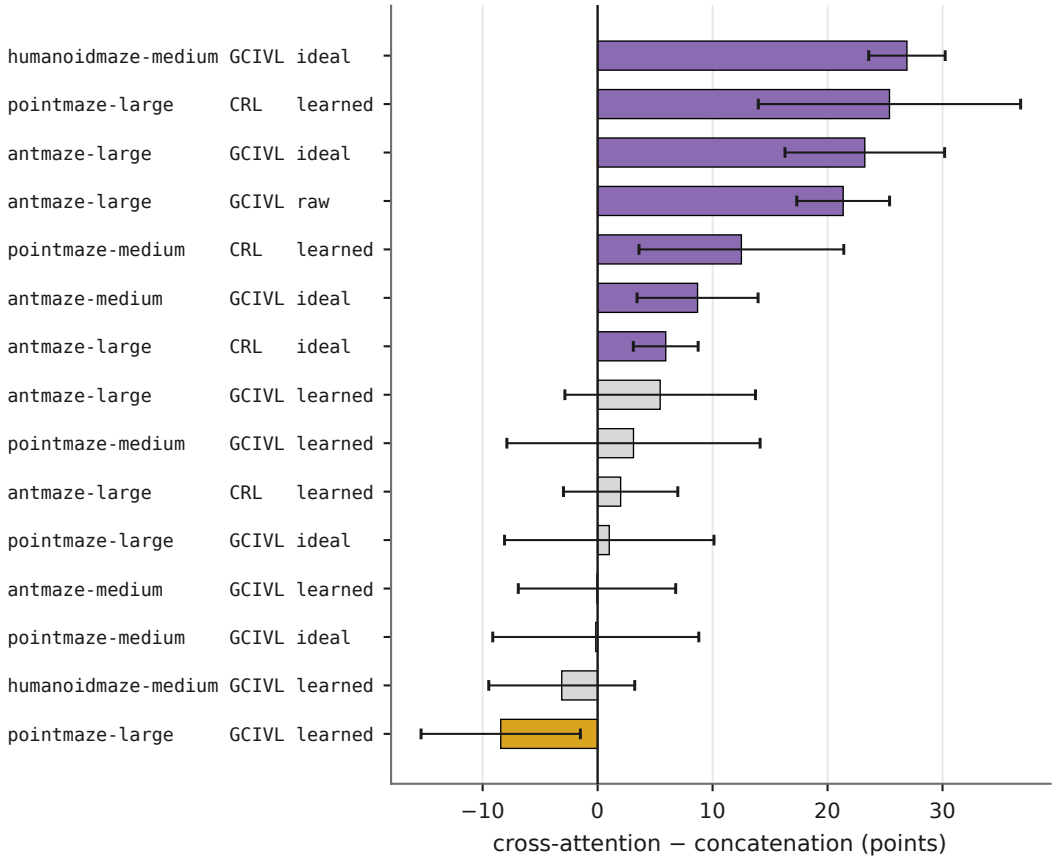


Figure 11: **Full interface sweep.** Cross-attention minus concatenation, with everything else held fixed within each row. Whiskers are 95% Welch intervals. Violet marks intervals excluding zero in favour of cross-attention, gold in favour of concatenation, grey a tie.

length at the attention peak is 9.0 and 11.8 cells, against 5.4 cells at the true argmin and 11.3 cells for a random landmark. The policy is attending to the goal region, not computing shortest paths through it.

I.3 Cross-Attention and the State Code Overlap

Cross-attention builds its query from the agent's state, so it introduces a state-dependent positional encoding implicitly. That predicts an overlap with the explicit state code, and we see one. Cross-attention is worth $+23.2$ $[+15.6, +30.9]$ on antmaze-large on its own, and $+33.2$ $[+25.2, +41.1]$ when extended to the critic, but only $+8.9$ $[-3.4, +21.2]$ once a state code is already present (Table 6). The two are largely different routes to the same quantity.

J Where the Interventions Do Not Help

Pointmaze. None of the interventions help. Cross-attention with the ideal representation is worth -0.2 $[-12.1, +11.8]$ on pointmaze-medium and $+1.0$ $[-9.7, +11.8]$ on pointmaze-large. Raising the landmark count from 64 to 256, the width used on antmaze, does not change that: the gains become $+1.8$ $[-7.2, +10.8]$ and -2.2 $[-9.9, +5.5]$. Pointmaze observations are four dimensional and the agent's coordinates are most of them, so there is little for a position encoding to disambiguate.

Manipulation. The interface result does not transfer either. Figure 12 reports GCIVL with the learned representation on the two manipulation tasks we ran. Cross-attention is significantly worse than concatenation on cube-single-play and no better on scene-play. These tasks have no maze

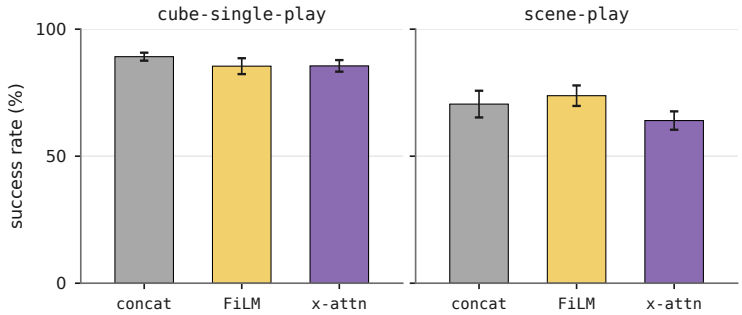


Figure 12: **Interfaces on manipulation tasks.** GCIVL with the learned representation. Cross-attention is worse than concatenation on cube-single-play, -3.6 $[-6.5, -0.7]$, and no better on scene-play, -6.5 $[-13.2, +0.2]$.

geometry and no single pair of coordinates that locates the agent, which is the structure the interventions exploit.