Direction-Conditioned Policies via Compositional Subgoal Scoring for Online Goal-Conditioned Reinforcement Learning

Swaminathan S K [1] **Damiya Gondha** [1 *] **Theyanesh Eswaramoorthy Rajahkrishnan** [2 *] **Aritra Hazra** [1]

Abstract

Hamilton–Jacobi–Bellman theory implies that the optimal goal-conditioned action depends on the goal only through the gradient of the goal-reaching distance at the current state, yet standard online GCRL still conditions the actor on the raw goal – a signal that is geometrically uninformative when the goal is far from the data distribution. We propose *Direction-Conditioned Policies* (DCP), a fully online method that decomposes goal-reaching into two components sharing one InfoNCE representation ψ: a subgoal-scoring step that selects a visited state z_t aligned with the final goal g in ψ_g, and a direction-conditioned actor that consumes the unit direction $\mathbf{d}_t$ and magnitude r_t from $\psi(s_t)$ to $\psi(z_t)$. The two components train jointly, factor cleanly at deployment (subgoal scoring is removed, while direction conditioning remains with g in place of z_t), and admit independent modification at the same $(\mathbf{d}_t, r_t)$ interface. We prove three results. First, direction sufficiency under HJB: the optimal action under control-affine dynamics depends on the goal only through the value gradient. Second, a quantitative bound showing that, under mild conditions on the learned representation and assuming the scoring rule returns an on-path z_t, the actor's conditioning input at training and at deployment coincide up to representation error and geodesic slack. Third, a controllable-subspace characterization of when directional conditioning fails. Across nine environments spanning navigation and manipulation, DCP improves over Contrastive RL on most final metrics, with the largest gains on manipulation and obstacle-interaction tasks; a qualitative analysis of the learned ψ-distance landscape shows the contrastive representation behaves as an online quasimetric encoding environment topology, and the single failure case (AntSoccer) localizes to a learned-gradient pathology that the theory anticipates.[1]

1. Introduction

Simulation is a scalable alternative to teleoperation for robot manipulation, enabling millions of parallel rollouts with contact-rich dynamics and no human operator. The challenge is sparse rewards: when goals are distant, the actor receives little direct signal. Standard Goal-Conditioned RL (GCRL) conditions the actor on the goal directly (Kaelbling, 1993). Contrastive RL (CRL) (Eysenbach et al., 2022) learns a representation ψ via InfoNCE such that $\langle \psi(s,a), \psi(g) \rangle$ estimates log-reachability of g (goal) from (s,a). However, the actor still receives the raw goal coordinate as input. This signal is geometrically uninformative when g is far from the data distribution: early in training $\psi_g(g)$ for a distant g is poorly calibrated, so any direction computed from $\psi_g(g)$ is unstable. We argue that the appropriate conditioning input is not the goal coordinate itself but a *direction* in the learned representation, and that this direction should be computed from a *visited* state z_t whose encoding the contrastive objective has already trained on.

Direction-Conditioned Policies (DCP) maintains a pool of visited states, selects z_t from the pool by an inner-product scoring rule that maximizes alignment with $\psi_g(g)$, and conditions the actor on the unit direction $\mathbf{d}_t$ from $\psi(s_t)$ to $\psi(z_t)$ together with a scalar magnitude r_t. Because z_t is selected to be aligned with g under ψ_g, the unit direction toward z_t approximates the unit direction toward g in ψ-space. At deployment, after ψ has been trained, the same unit direction is computed with g in place of z_t and the pool is no longer needed. We also evaluate a simpler variant, *Self-Subgoal Conditioning* (SSGC), which uses the same scoring but conditions on the raw subgoal coordinate z_t instead of the direction; this ablation isolates the contribution of directional abstraction. No teleoperation, demonstrations, or

[*]Equal contribution [1]Department of Computer Science and Engineering, Indian Institute of Technology Kharagpur, Kharagpur, India [2]Department of Mechanical Engineering, Indian Institute of Technology Kharagpur, Kharagpur, India. Correspondence to: Swaminathan S K <swami2004@kgpian.iitkgp.ac.in>.

Accepted to the 2nd Workshop on Compositional Learning at ICML 2026, Seoul, South Korea. Copyright 2026 by the author(s).

[1]Code and reproduction scripts: https://anonymous.4open.science/r/dcp-supplement-anon-CF41/

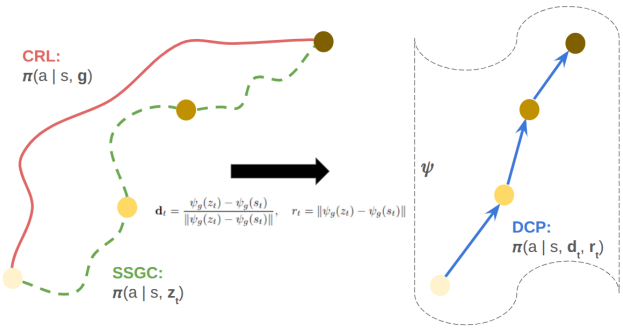


Figure 1. Conditioning Strategies. CRL: final goal g. SSGC: subgoal z_t. DCP: direction $\mathbf{d}_t$ and magnitude r_t in ψ-space.

offline data are required.

DCP decomposes goal-reaching into two pieces sharing one InfoNCE representation: scoring selects a visited state aligned with the goal in ψ-space, and a direction-conditioned actor consumes the unit direction from the current state toward that selected state. The two pieces train jointly but factor cleanly at deployment, where the pool is no longer needed and the actor is fed the unit direction toward g directly. We give this design a principled foundation in Section 4: the Hamilton–Jacobi–Bellman (HJB) equation identifies the unit gradient of the goal-reaching distance as the minimal sufficient statistic for the optimal action under control-affine dynamics. DCP is the online realization of this fact through CRL's InfoNCE-learned quasimetric. In heuristic-search vocabulary, ψ is a learned consistent quasimetric and the actor performs greedy descent on it in continuous time.

The primary contributions of this work are as follows.

- We propose a direction-conditioned actor for online GCRL using $[\mathbf{d}_t, r_t]$ in ψ-space.
- We introduce a subgoal scoring rule that selects visited states aligning with the goal in ψ_g.
- We establish three theoretical results: direction sufficiency under HJB, planning invariance at the conditioning interface, and a controllable-subspace failure characterization.
- We evaluate the proposed method across nine environments, showing five-seed gains over Contrastive RL on most final metrics, with AntSoccer failure consistent with the regime anticipated by theory.

2. Related Work

DCP connects several lines of prior work: online contrastive goal-reaching, HJB/quasimetric formulations of control, and compositional subgoal methods. The contribution is an online actor-conditioning scheme that consumes the HJB-direction-sufficient statistic, enforces planning invariance at the conditioning interface, and remains orthogonal to representation-improvement work within Contrastive RL.

HJB-based and Quasimetric Methods for Goal-reaching. The Hamilton–Jacobi–Bellman equation (Bardi & Capuzzo-Dolcetta, 1997; Bertsekas et al., 2010) characterizes the optimal action via the gradient of the cost-to-go (Theorem 1). Wang et al. (2023) learn an explicit quasimetric for offline goal-reaching (QRL); Giammarino & Qureshi (2025) impose the Eikonal PDE constraint on a hierarchical quasimetric (Eik-QRL) and explicitly use the HJB form for value learning. Both target value learning rather than actor conditioning. Myers et al. (2024) show that contrastive successor features can be reparameterized to satisfy the triangle inequality, providing the consistency property (in the heuristic-search sense). DCP is, to our knowledge, the first online actor-conditioning method that consumes the gradient of this learned consistent quasimetric; the orthogonality to prior value-learning work in this area is by construction.

Dual Goal Representations and Planning Invariance. Park et al. (2025b) prove that the dual representation $\phi^{\vee}(g) := s \mapsto d^*(s, g)$ is sufficient to recover the optimal goal-conditioned policy and is invariant to exogenous goal-observation noise; the DCP conditioning vector $\mathbf{d}_t$ is the local gradient of this dual representation at the current state, inheriting noise invariance under the same identification. Myers et al. (2025a) define planning invariance – a policy taking the same action whether conditioned on an on-path waypoint or the final goal – and prove that some learned representations have it as an emergent property, with implications for horizon generalization. Theorem 2 establishes a related statement *at the actor's interface*: when the learned d_ψ approximates d^* uniformly to within δ and the scoring rule returns an ε-on-path z, the actor's conditioning input at training (toward z) and at deployment (toward g) coincide up to $O(\sqrt{\delta} + \sqrt{\varepsilon})$. Planning invariance is thereby inherited at the conditioning interface from the representation's approximation properties, without an explicit hierarchy. HILP (Park et al., 2024b) similarly conditions on Hilbert-metric directions but requires offline metric supervision and a symmetric metric; CRL's quasimetric is non-symmetric and DCP recovers the directional conditioning online via InfoNCE.

Online Contrastive RL. CRL (Eysenbach et al., 2022) is our baseline: an InfoNCE critic paired with SAC. Bortkiewicz et al. (2024) establish CRL as the reference online GCRL method and provide the JAX infrastructure we build on. Scaled CRL (Wang et al., 2025) improves via depth scaling (1000-layer networks); E-CRL (Tangri et al., 2025) adds equivariance; both improve the *representation*, while DCP improves the *actor's conditioning interface* given any representation. The two contributions are orthog-

onal and can be integrated. Our empirical comparison is therefore a controlled CRL-backbone interface study: critic, replay, optimizer, training budget, and implementation are held fixed while the actor-conditioning signal is changed. Liu et al. (2024) show that online contrastive RL with a single fixed goal can elicit useful skills and exploration without any subgoal machinery, contrasting with our explicit subgoal scoring. Park et al. (2025a) argue that reducing the effective planning horizon is a key lever for scaling RL; we view this as complementary, since we modify the actor's conditioning input rather than the training-time horizon itself.

Embedding-difference Conditioning. Plan Arithmetic / Compositional Plan Vectors (CPV) (Devin et al., 2019) conditions a policy on differences in learned plan embeddings, enabling arithmetic composition over task or trajectory-level representations. DCP shares the high-level idea that a difference vector in representation space can be more useful than raw goal coordinates. The operating regime is different: CPV is one-shot imitation over reference/partial trajectories for explicit skill compositions, whereas DCP is online GCRL with final goals and no demonstrations. Applying CPV directly would therefore require changing the evaluation setting by adding reference trajectories and explicit task-composition structure.

Subgoal and Landmark Planning. DCP's visited-state scorer is related to subgoal and landmark methods such as L3P / World Model as a Graph (Zhang et al., 2020), which learns latent landmarks and plans over a reachability graph. The shared intuition is that intermediate states can make long-horizon sparse goal-reaching easier. The distinction is architectural: L3P couples a goal-conditioned Q-learning controller with persistent landmark nodes and deployment-time graph search. DCP uses subgoals only as a training-time scaffold; at deployment there is no landmark graph, planner, or waypoint pool. Thus L3P is related landmark-planning work, while the experiments here isolate the effect of changing the actor-conditioning interface.

Compositional and Hierarchical RL. A line of recent work decomposes long-horizon goal-reaching into reusable sub-units. Yalcinkaya et al. (2024) encode tasks as automata embeddings; Liang et al. (2023) learn interpretable skill abstractions via diffusion; Bakirtzis et al. (2024) formalize compositional RL through a categorical lens. Most of these methods rely on an explicit two-level architecture (separate high- and low-level policies, often trained at different rates) or external task structure. DCP is *flat*: a single direction-conditioned actor plus a scoring rule, sharing one ψ trained jointly with InfoNCE. The two components share one interface ($\mathbf{d}_t, r_t$ in ψ-space) and can be modified independently behind it – scoring selects z_t, direction conditioning consumes the resulting unit vector – without explicit hierarchical structure.

Offline Goal-conditioned RL. HIQL (Park et al., 2023), Myers et al. (2025b), and Zheng et al. (2025) learn from pre-collected datasets, exploiting offline metric supervision DCP does not require. Park et al. (2024a) introduce OGBench for benchmarking such methods. We do not run head-to-head comparisons; the operating regime (online with no demonstrations) differs.

3. Proposed Method

3.1. Contrastive Representation

Following CRL (Eysenbach et al., 2022), we train a state-action encoder ψ_{sa} and goal encoder ψ_g via InfoNCE:

$$\mathcal{L} = -\mathbb{E}\left[\log \frac{e^{\langle \psi_{sa}(s,a),\, \psi_g(g)\rangle}}{\sum_{g'} e^{\langle \psi_{sa}(s,a),\, \psi_g(g')\rangle}}\right] \tag{1}$$

After training, $\langle \psi_{sa}(s,a), \psi_g(g)\rangle$ is large when (s,a) tends to lead to g and small otherwise.

3.2. Subgoal Pool and Scoring (SSGC & DCP)

Both methods maintain a fixed-size pool $\mathcal{P}$ of recently visited states, updated online. At each step a subgoal $z_t \in \mathcal{P}$ is selected by:

$$z_t = \arg\max_{z \in \mathcal{P}} \langle \psi_g(z), \psi_g(g)\rangle \tag{2}$$

The dot product selects pool states whose ψ_g-encoding is most similar to $\psi_g(g)$ under the InfoNCE inner product. Conceptually, these are states the contrastive objective has placed in the same region of ψ-space as the goal; the resulting unit direction toward z_t is therefore close to the unit direction toward g in ψ-space, while z_t itself was actually visited and so is on the support of the goal encoder's training distribution.

SSGC conditions the actor on $[s_t, z_t]$: a concrete, achievable waypoint instead of the distant goal. **DCP** instead computes

$$\mathbf{d}_t = \frac{\psi_g(z_t) - \psi_g(s_t)}{\|\psi_g(z_t) - \psi_g(s_t)\|}, \quad r_t = \|\psi_g(z_t) - \psi_g(s_t)\| \tag{3}$$

and conditioning the actor on $[s_t, \mathbf{d}_t, r_t]$. Here, $\mathbf{d}_t \in \mathbb{S}^{d-1}$ is a unit vector in representation space pointing from s_t toward z_t, and r_t encodes the magnitude of the conditioning input in ψ-space. Algorithm 1 lists the rollout step and the actor and critic losses in pseudocode.

Both methods are trained end-to-end with SAC (Haarnoja et al., 2018) alongside the contrastive objective (Eq. 1), with

Algorithm 1 DCP: rollout step, subgoal scoring, and losses.

Require: goal g, visited-state pool $\mathcal{P}$, encoders ψ_{sa}, ψ_g, policy π

Function DIRECTION(ψ_z, ψ_s)
  $\mathbf{d}_t \leftarrow (\psi_z - \psi_s)/\|\psi_z - \psi_s\|; \quad r_t \leftarrow \|\psi_z - \psi_s\|$
  return $(\mathbf{d}_t, r_t)$

Function SELECTSUBGOAL$(g, \mathcal{P})$ {every 25 env-steps, 32 of 512}
  $\mathcal{C} \leftarrow$ SAMPLEWITHOUTREPLACEMENT$(\mathcal{P}, 32)$
  return $\arg\max_{z \in \mathcal{C}} \langle \psi_g(z),\, \psi_g(g) \rangle$

Function ROLLOUTSTEP(s, z) {z stored in buffer goal slot}
  $(\mathbf{d}_t, r_t) \leftarrow$ DIRECTION$(\psi_g(z),\, \psi_g(s[\text{goal dims}]))$
  return $\pi(s, \mathbf{d}_t, r_t)$

Function CRITICLOSS(s, a, g') {backward InfoNCE over (s, a)}
  $\phi \leftarrow \psi_{sa}(s, a); \quad \psi \leftarrow \psi_g(g')$
  $\ell_{ij} \leftarrow -\|\phi_i - \psi_j\|$ {negative-L2 energy}
  return $-\operatorname{mean}_i \left[\ell_{ii} - \operatorname{logsumexp}_j \ell_{ji}\right]$

Function ACTORLOSS(s, z) {encoders `stop_grad`; SAC entropy added}
  $\psi_z \leftarrow \psi_g(z); \quad \psi_s \leftarrow \psi_g(s[\text{goal dims}])$
  $(\mathbf{d}_t, r_t) \leftarrow$ DIRECTION(ψ_z, ψ_s); $a \leftarrow \pi$.SAMPLE$(s, \mathbf{d}_t, r_t)$
  return $\operatorname{mean} \|\psi_{sa}(s, a) - \psi_z\| + \alpha \log \pi(a|s, \mathbf{d}_t, r_t)$

no changes to the critic or replay buffer. The actor objective remains within the same CRL/JaxGCRL training stack; DCP changes the conditioning object passed to the actor and the positive target induced by the selected waypoint, not the critic, data collection, optimizer, or schedule. During *deployment*, the subgoal pool is not needed since $\mathbf{d}_t$ is computed directly as $(\psi_g(g) - \psi_g(s_t))/\,\|\psi_g(g) - \psi_g(s_t)\|$; therefore DCP is deployed as a standard goal-conditioned policy. ψ_g is the same goal encoder CRL already trains and DCP incurs no extra training cost or deployment complexity.

3.3. Rationale behind Scoring and Direction

Why score subgoals from a visited pool? The conceptually simplest version of direction conditioning would feed the actor $\mathbf{d}_t = (\psi_g(g) - \psi_g(s_t))/\,\|\psi_g(g) - \psi_g(s_t)\|$ directly. We avoid this early in training, because the encoded goal direction is not yet informative: ψ has not been shaped by the InfoNCE objective enough for $\psi_g(g)$ to be meaningfully separated from arbitrary visited states in ψ-space. In this regime $\langle \psi_g(g), \psi_{sa}(s, a) \rangle$ is close to chance and the unit vector $\mathbf{d}_t$ is unstable. Conditioning on a miscalibrated direction is worse than conditioning on the raw goal coordinate (see SSGC and CRL ablations in Sec. 5.3).

Subgoal scoring mitigates this issue. Equation (2) selects z_t as the visited state whose $\psi_g(z_t)$ is most aligned with $\psi_g(g)$ under the InfoNCE inner product, restricting attention to candidates the contrastive objective has already separated from the rest of the buffer. The chosen z_t is therefore a state for which the direction $\psi_g(z_t) - \psi_g(s_t)$ is computed from *two* encodings the contrastive objective has trained on, rather than one trained encoding ($\psi_g(s_t)$) and one poorly calibrated encoding ($\psi_g(g)$ for far-from-data g). At deployment, after InfoNCE training has shaped ψ, the same construction with z_t replaced by g becomes calibrated and the pool is no longer needed.

Why condition on a direction in ψ-space rather than raw z_t? SSGC conditions on z_t directly. This can be effective during training but degrades at deployment, because the actor sees the distribution of subgoal coordinates during training and the distribution of *final* goal coordinates at test time, and the two need not coincide – a train–test mismatch documented in our SSGC results. DCP avoids this by mapping both training-time and deployment-time conditioning into the same object: the unit direction $\mathbf{d}_t$ in ψ-space. Because scoring selects z_t to be aligned with g in ψ_g, the unit direction toward z_t approximates the unit direction toward g, and hence at deployment, we can compute the same unit direction with g in place of z_t.

Theorem 2 formalizes this as planning invariance at the conditioning interface. HILP (Park et al., 2024b) learns a Hilbert metric offline and conditions on directions in that metric; DCP recovers an analogous directional conditioning *online* via the InfoNCE quasimetric of CRL, without offline metric supervision. CRL's quasimetric is non-symmetric (Myers et al., 2024), which lets DCP express asymmetric goal-reaching geometry that a symmetric Hilbert embedding cannot.

4. Theoretical Formulation: Direction Conditioning under HJB

We work in continuous time on $\mathcal{S} \subseteq \mathbb{R}^n$, $\mathcal{A} \subseteq \mathbb{R}^m$, with deterministic locally-Lipschitz dynamics $\dot{s} = f(s, a)$ and lower-semicontinuous cost $c \geq 0$. The optimal goal-reaching cost-to-go is $d^*(s, g) := \inf_{a(\cdot)} \int_0^{T_g} c\,\mathrm{d}t$ subject to $s(0) = s$, $s(T_g) = g$, and we assume $d^*(\cdot, g)$ is C^1 on $\mathcal{S} \setminus \{g\}$ (the standard regularity hypothesis under which the HJB equation has classical solutions; the viscosity-solution extension is well-known (Bardi & Capuzzo-Dolcetta, 1997)). The learned representation $\psi : \mathcal{S} \to \mathbb{R}^k$ induces a distance $d_\psi(s, g) := \|\psi(g) - \psi(s)\|$, and DCP feeds the actor the unit direction $\mathbf{d}_t = -\nabla_s d_\psi / \|\nabla_s d_\psi\|$ and magnitude $r_t = d_\psi$. The implementation uses the equivalent ψ-space

form $\mathbf{d}_t = (\psi(g) - \psi(s))/\|\cdot\|$; the two are related by the pullback $\nabla_s d_\psi = -J_\psi(s)^\top \widehat{\psi(g) - \psi}(s)$ and encode the same information when J_ψ has full rank. Full assumptions and notation are in Appendix A.1.

Section 4 states the three theorems that justify DCP's design; full proofs and remarks are deferred to Appendix A.

Theorem 1 (Direction Sufficiency under HJB). *Under the regularity above, the optimal goal-conditioned feedback law $a^*(s, g)$ depends on g only through $\nabla_s d^*(s, g)$: there is a function $\bar{a} : \mathcal{S} \times \mathbb{R}^n \to \mathcal{A}$ such that $a^*(s, g) = \bar{a}(s, \nabla_s d^*(s, g))$. If, in addition, the dynamics are control-affine with $f(s, a) = f_0(s) + G(s)a$ and $c(s, a) = c_0(s) + \frac{1}{2}a^\top R(s)a$, $R(s) \succ 0$, then $\bar{a}$ is* linear *in its second argument and $a^*(s, g) = -R(s)^{-1}G(s)^\top \nabla_s d^*(s, g)$. The unit direction $\widehat{\nabla_s d^*}$ sets the orientation of a^* and the magnitude $\|\nabla_s d^*\|$ sets its scale.*

Intuition: the HJB equation $\inf_a[c(s, a) + \nabla_s d^* \cdot f(s, a)] = 0$ couples g to the optimization only through the gradient term, and minimizing over a yields a feedback law that is a function of $(s, \nabla_s d^*)$ alone. DCP's conditioning interface $(\mathbf{d}_t, r_t)$ is precisely this minimal sufficient statistic with d^* replaced by the learned d_ψ. The full proof, the connection to Pontryagin's principle, and the viscosity-solution extension are in Appendix A.2.

Theorem 2 (Planning Invariance at Conditioning Interface). *Assume* driftless *control-affine dynamics $f(s, a) = G(s)a$ with quadratic cost (B0); a state z that is ε-*on-path *from s to g in the sense $d^*(s, g) \le d^*(s, z) + d^*(z, g) \le d^*(s, g) + \varepsilon$ (B1); a learned d_ψ approximating d^* uniformly to within δ with L-Lipschitz gradients in both arguments (B2); and $\|\nabla_s d^*(s, g)\| \ge m > 0$ (B3). Let $\mathbf{d}_t^{(z)} = -\widehat{\nabla_s d_\psi}(s, z)$ and $\mathbf{d}_t^{(g)} = -\widehat{\nabla_s d_\psi}(s, g)$. Then*

$$\left\|\mathbf{d}_t^{(z)} - \mathbf{d}_t^{(g)}\right\| \le \frac{8\sqrt{2L\delta}}{m} + \frac{2\sqrt{2L\varepsilon}}{m}. \tag{4}$$

Intuition: under (B0) the optimal trajectories of $d^*(\cdot, g)$ and $d^*(\cdot, z)$ share a common geodesic tangent at s when z is on-path, so the unit gradients agree to $O(\sqrt{\varepsilon})$; the Glaeser-type $L^\infty \to W^{1,\infty}$ inequality converts δ-uniform value approximation into $O(\sqrt{\delta})$ gradient perturbation, which the unit-vector stability lemma (using B3) carries to the unit direction. Consequently, for a K_π^d-Lipschitz policy, the actor's training-time and deployment-time outputs agree up to $O(\sqrt{\delta} + \sqrt{\varepsilon})$. The statement is conditional on the on-path assumption: the inner-product scorer is not claimed to guarantee (B1) for arbitrary learned representations, and Appendix A.3 states this explicitly. The driftless hypothesis is used only for the geodesic-colinearity step; Appendix 5 details which parts remain valid with drift. Full proof, drift-extension discussion, and the connection to dual goal representations (Park et al., 2025b) and horizon generalisation (Myers et al., 2025a) are in Appendix A.3.

Table 1. Final DCP vs. CRL comparison across nine environments (five seeds each).

Environment	Task	Time Near Goal	Success ≥1
AntMaze Big	Navigation	↑ DCP	↑ DCP
AntMaze Hardest	Navigation	↑ DCP	↑ DCP
Ant U-Maze	Navigation	↓ CRL	↑ DCP
Humanoid U-Maze	Navigation	↑ DCP	↑ DCP
AntPush	Nav + Manip	↑ DCP	≈
PusherEasy	Manipulation	↑ DCP	↑ DCP
PusherHard	Manipulation	↑ DCP	↑ DCP
PusherHard Far	Manipulation	↑ DCP	↑ DCP
AntSoccer	Manipulation	↓ CRL	↓ CRL

↑/↓ mark the higher final mean; ≈ marks a small or uncertainty-overlapped difference. Exact checkpointed means and 95% confidence intervals are in Appendix Tables 4–5. This table is an effect-size summary, not a significance test.

Theorem 3 (Failure Mode: Uncontrollable Goal Subspace). *Under control-affine $f(s, a) = f_0(s) + G(s)a$, define the controllable subspace $\mathcal{C}(s) := \mathrm{im}(G(s))$ and the controllable gradient component $\nabla_s^{\mathcal{C}} d^* := \Pi_{\mathcal{C}(s)} \nabla_s d^*$. Then (i) a^* depends on $\nabla_s d^*$ only through $\nabla_s^{\mathcal{C}} d^*$; (ii) if $\|\nabla_s^{\mathcal{C}} d^*(s, g)\| \le \rho \|\nabla_s d^*(s, g)\|$ for $\rho \ll 1$, then under quadratic cost $\|a^*(s, g)\| \le \rho \|R^{-1}\|_{\mathrm{op}} \|G\|_{\mathrm{op}} \|\nabla_s d^*\|$ and the directional gradient signal $\mathbf{d}_t$ is uninformative; (iii) on a state z for which $\|\nabla_s^{\mathcal{C}} d^*(s, z)\| \ge \Theta(\|\nabla_s d^*(s, z)\|)$, $\mathbf{d}_t(s, z)$ is informative and the actor can make progress toward z.*

Intuition: $G(s)^\top$ annihilates $\mathcal{C}(s)^\perp$, so only the controllable component of the gradient enters the action selection; when this component is small, every action with $G(s)a \perp \nabla_s^{\mathcal{C}} d^*$ is approximately optimal and the directional signal carries no usable information. The full proof and the AntSoccer instantiation (where the failure is on the *learned* gradient $\nabla_s d_\psi$ rather than $\nabla_s d^*$, due to a narrow visited pool that does not expose ball-relevant directions to InfoNCE) are in Appendix A.4 and A.5.

Compositional Reading. In heuristic-search vocabulary, ψ is a learned consistent quasimetric (consistency from the contrastive-successor-feature triangle inequality of Myers et al. (2024)); $\mathbf{d}_t$ is the negative gradient of this quasimetric; the actor performs greedy best-first descent in continuous time. The waypoint pool plays the role of an open list during training; at deployment, once ψ has been trained, the list collapses to $\{g\}$ and the policy becomes a standard goal-conditioned controller. We frame this as an analogy motivating the algorithm rather than a formal claim: full A*-style admissibility ($d_\psi \le d^*$ pointwise) is not implied by InfoNCE training and is not required; the formal claims are Theorems 1–3.

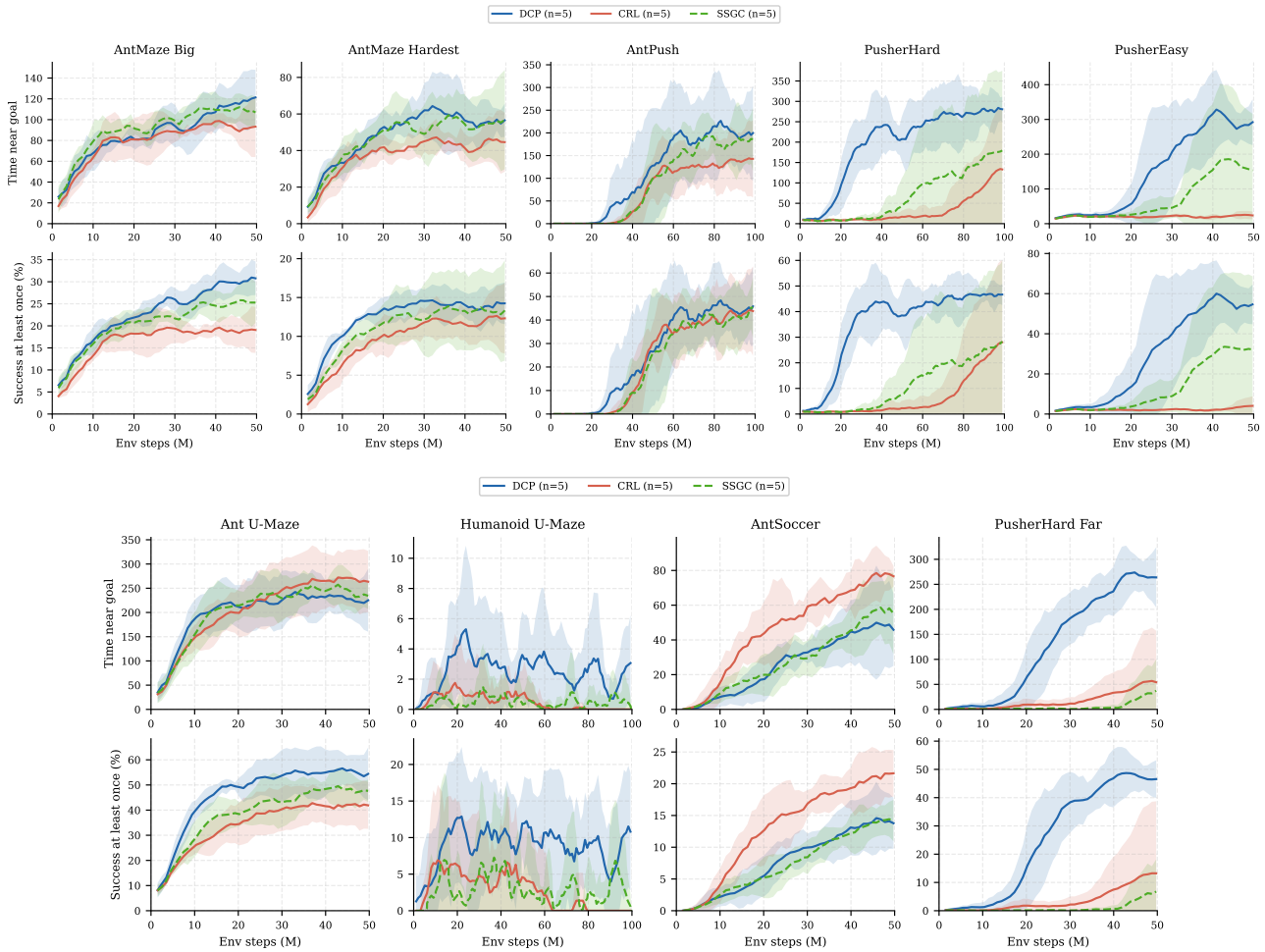


Figure 2. Learning curves for time near goal (top of each strip) and success at least once (bottom) across all nine environments. Lines show five-seed means and shaded bands show 95% confidence intervals over seeds. AntPush, PusherHard, and Humanoid U-Maze use 100M-step budgets; the remaining environments use 50M-step budgets. Final five-seed values are summarized in Table 1, with checkpointed means and confidence intervals in Appendix Tables 4–5.

5. Experimental Results

5.1. Setup and Environment Details

We evaluate across nine environment configurations using Brax (Freeman et al., 2021). Unless noted, runs use 512 parallel environments, 50M env steps, episode length 1000, unroll length 62, batch size 256, replay buffer size 10 000, discount γ=0.99, and learning rate 3×10^{-4}.

AntPush, PusherHard, and Humanoid U-Maze are trained for 100M steps; Humanoid U-Maze uses 128 parallel environments and batch size 128 due to compute constraints. The subgoal pool holds 512 states; 32 candidates are scored every 25 steps. We use backward InfoNCE as the contrastive loss and negative L2 norm as the energy function, following the official JaxGCRL run scripts (Bortkiewicz et al., 2024); CRL numbers may therefore differ from those reported in the JaxGCRL paper, which uses symmetric InfoNCE. Primary experiments use five seeds; Figure 2 reports learning curves with mean and 95% confidence intervals over seeds, and Appendix Tables 4–5 give checkpointed values including the final checkpoint. The baseline throughout the main comparison is CRL (Eysenbach et al., 2022) with identical architecture, hyperparameters, and training budget; the only difference is actor conditioning. We therefore interpret the results as a controlled interface comparison within one strong online GCRL backbone, rather than as an exhaustive benchmark over all goal-conditioned RL families.

- *AntMaze Big / Hardest*: a quadruped navigates increasingly complex mazes to reach a sparse goal.
- *Humanoid U-Maze*: a 17-DOF humanoid navigates a U-shaped maze; the same task structure as AntMaze but with a far higher-dimensional body (268 state dimensions vs. 29).
- *PusherEasy / PusherHard*: a 7-DOF arm pushes a

cylinder to a goal XY position; Easy and Hard differ in goal-distribution difficulty.
- *AntPush*: a quadruped navigates a room with a movable box obstacle to reach a goal XY position encoding the ant's own position.
- *AntSoccer*: a quadruped must kick a ball to a target position; the goal is defined on the ball's XY position.

We report two metrics, both using the same goal-region threshold:

- *Time near goal* (primary): average timesteps per episode within the goal region, measuring sustained proximity.
- *Success at least once*: fraction of episodes entering the goal region at least once, measuring reaching ability.

5.2. Qualitative Evaluation: Quasimetric Landscape

Figure 3(a) renders $\|\psi(x, y) - \psi(g)\|$ over AntMaze Big at convergence: DCP's actor consumes the gradient of this field; CRL receives only the raw goal coordinate. We show a seed on which both methods reach the goal, so the comparison is about ψ geometry rather than outcomes. Two structural claims are visible. First, contours bend around walls in both panels – InfoNCE produces an online quasimetric encoding reachability through corridors (Wang et al., 2023; Myers et al., 2024), the same structure HILP (Park et al., 2024b) obtains offline via explicit Hilbert-metric supervision. Second, DCP's basin is tight along the reachable corridor while CRL's is more diffuse, consistent with the joint-training coupling: DCP's actor reads ψ's gradient at every step, so any ψ drift away from a tight reachable basin directly hurts the policy gradient and is corrected during training; CRL's actor is ψ-independent and so does not impose this constraint. App. Fig. 4 shows the same comparison on a seed where CRL fails; the diffuse-basin pattern is more pronounced there.

5.3. Empirical Performance

Navigation – AntMaze and HumanoidMaze. DCP improves reaching across the navigation tasks and is especially useful as the body dynamics become harder. On the AntMaze variants, DCP generally improves success-at-least-once over CRL, while time near goal is similar or better except for the final Ant U-Maze checkpoint. Humanoid U-Maze is difficult for all methods, but DCP is the only method with consistent nonzero progress, suggesting that directional conditioning remains useful even when the same maze structure is paired with a much higher-dimensional body. SSGC is competitive on the lower-dimensional mazes but does not scale as cleanly to Humanoid, where raw subgoal coordinates create a harder conditioning problem than directions in ψ-space.

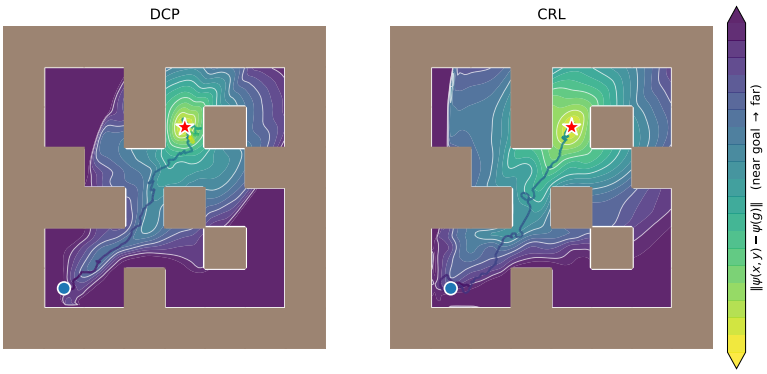
Object Manipulation – Pusher and AntPush. DCP has its clearest gains on tasks that require continuous corrective control around contact or obstacles. On the Pusher family, DCP improves both reaching and sustained proximity, while SSGC often lags CRL, indicating that raw subgoal conditioning is insufficient. AntPush also favors DCP on sustained occupancy: the time-near-goal gap opens during training and remains visibly higher for DCP across the later part of the 100M-step run, while success-at-least-once is comparable at the final checkpoint. Thus the AntPush gain is best read as a sustained-occupancy improvement rather than a first-contact improvement.

Goal Geometry Ablation. CRL performs worse on PusherHard than in the original JaxGCRL paper (Bortkiewicz et al., 2024). We traced this to a goal distribution shift across JAX/Brax versions and found that our PRNG biases goals toward the near arc (Fig. 3(b)), requiring repositioning behind the object. Scaled CRL (Wang et al., 2025) documents similar discrepancies. To verify, we created *PusherHard-Far*, constraining goals to the far arc. CRL improves relative to the harder near-arc setting, but DCP still leads on the final metrics, confirming robustness to goal geometry.

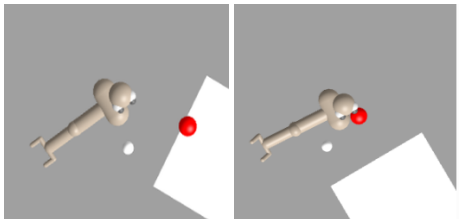
Failure Case – AntSoccer. AntSoccer is a failure case for DCP, with CRL $>$ DCP $>$ SSGC throughout training. The mechanism is analysed in Section A.5: the goal coordinate is the ball's position, and the visited pool early in training rarely contains states with displaced balls, so the InfoNCE objective receives little signal that distinguishes ball-position changes from ant-position changes. Empirically $\nabla_s d_\psi(s, g_{\text{ball}})$ lies in the controllable subspace (so the actor receives a non-trivial signal) but its projection onto the directions that actually reduce $d^*(s, g_{\text{ball}})$ is small. CRL is less exposed to this learned-gradient misalignment because its actor receives the raw goal coordinate, while the InfoNCE loss still provides a weaker but non-zero action-value signal through the SAC critic.

5.4. Summary and Discussion

DCP's advantage grows with task difficulty (Table 1): the clearest gains appear on manipulation and obstacle-interaction tasks such as Pusher and AntPush, and on high-dimensional navigation such as Humanoid U-Maze. SSGC's underperformance on several manipulation tasks isolates the source: subgoal *scoring* can help, but conditioning on raw subgoal coordinates creates a train–eval coordinate-distribution mismatch; DCP avoids this by mapping both training-time and deployment-time conditioning into the same ψ-space direction. DCP carries no deployment overhead beyond CRL: at test time $\mathbf{d}_t$ is computed directly from g, with no pool or candidate scoring – an advantage over methods requiring waypoint graphs or planner modules at



(a) ψ-distance landscape



(b) PusherHard goal geometry

Figure 3. **(a)** Iso-distance contours of $\|\psi(x, y) - \psi(g)\|$ over AntMaze Big under DCP (left) and CRL (right) on a seed where both reach the goal; contours bend around walls in both panels (online quasimetric), and DCP's basin is sharper along the start–goal corridor. **(b)** PusherHard goal geometry (red: goal; white: cylinder). Left: PusherHard-Far (far arc, natural push). Right: default (near arc, requires repositioning behind the cylinder).

Table 2. Periodic-burst Zero-shot Deployment (P=100, L=10). Cumulative reach (%) over 1000 steps, 3 seeds; bold Δ exceeds 1σ. Final col is $\Delta_{\text{DCP}} - \Delta_{\text{CRL}}$. Per-env saturation rates in Appendix C.

Env	DCP base	DCP Δ	CRL base	CRL Δ	DCP − CRL
Ant U-Maze	59.4 ± 8.8	$\mathbf{+10.4 \pm 7.8}$	44.8 ± 5.3	$\mathbf{+14.6 \pm 5.3}$	-4.2
AntMaze Big	33.3 ± 8.2	$\mathbf{+11.5 \pm 1.5}$	26.0 ± 2.9	$+5.2 \pm 3.9$	$+6.2$
AntMaze Hardest	15.6 ± 6.8	$+2.1 \pm 2.9$	19.8 ± 5.3	-5.2 ± 2.9	$+7.3$
AntPush	45.8 ± 1.5	$\mathbf{+22.9 \pm 8.2}$	43.8 ± 14.2	-7.3 ± 11.5	**+30.2**
PusherEasy	44.8 ± 3.9	$\mathbf{+20.8 \pm 3.9}$	1.0 ± 1.5	0.0 ± 0.0	**+20.8**
PusherHard	52.1 ± 2.9	$\mathbf{+12.5 \pm 5.1}$	19.8 ± 25.8	$+2.1 \pm 2.9$	**+10.4**
PusherHard Far	53.1 ± 2.6	-1.0 ± 7.8	8.3 ± 7.4	-1.0 ± 5.3	0.0
Humanoid U-Maze	6.2 ± 8.8	-1.0 ± 9.0	0.0 ± 0.0	0.0 ± 0.0	-1.0
AntSoccer	11.5 ± 3.9	$+4.2 \pm 8.2$	24.0 ± 5.3	-7.3 ± 3.9	**+11.5**
overall (paired)		$+9.1 \pm 10.6$		$+0.1 \pm 8.3$	**+9.0**
W/T/L		18/5/4		8/9/10	

test time.

5.5. Saturation-Aware Zero-Shot Deployment

The deployment-time policy can be improved *without retraining* by replacing the action with $\mathcal{U}[-1, 1]$ for $L = 10$ steps every $P = 100$ steps. We hypothesize this helps DCP because DCP recomputes $\mathbf{d}_t$ at every step from the current state – a noise-induced perturbation produces a fresh corrective input – whereas CRL reads only the static raw goal and has no analogous channel. Table 2 reports paired deployment results across the same tasks. The intervention improves DCP more consistently than CRL, with the largest differentials on AntPush and the Pusher tasks. Per-environment action-saturation rates (Appendix C) confirm the precondition that deterministic deployment leaves headroom but do not predict the magnitude of Δ; the CRL control indicates that the effect is tied to DCP's direction recomputation rather than to generic SAC action saturation.

6. Conclusion and Future Work

We presented Direction-Conditioned Policies (DCP), an online GCRL method that decomposes goal-reaching into a subgoal-scoring step and a direction-conditioned actor sharing one InfoNCE representation. The two components train jointly and factor cleanly at deployment: the pool is removed and the actor consumes the unit direction toward g directly, so DCP carries no test-time overhead beyond CRL. Empirically, DCP improves over CRL on most five-seed final metrics across nine Brax environments, with the largest gains on Pusher tasks, a clear sustained-proximity gain on AntPush, and nonzero progress on Humanoid U-Maze. The SSGC ablation isolates the source of the gain: subgoal *scoring* is valuable, but conditioning on raw subgoal coordinates hurts at deployment due to a train–eval coordinate-distribution mismatch that directional abstraction in ψ-space resolves.

Theoretically, DCP is the online realization of the Hamilton–Jacobi–Bellman direction-sufficiency principle (Theorem 1): under control-affine dynamics, the optimal action depends on the goal only through the gradient of the cost-to-go, and our planning-invariance bound (Theorem 2) shows that on-path subgoal scoring keeps the actor's training and deployment conditioning inputs within representation error and geodesic slack of each other. The AntSoccer failure case is consistent with the theory's controllable-subspace characterization (Theorem 3): the visited pool early in training rarely contains states with displaced balls, so the InfoNCE objective receives little signal that distinguishes ball-position from ant-position changes, and the learned gradient $\nabla_s d_\psi(s, g_{\text{ball}})$ is dominated by ant-state directions that do not reduce the true cost-to-goal. This identifies a concrete precondition for DCP: the visited pool must expose the goal-relevant directions to InfoNCE for the learned direction to

be useful.

Several directions follow. First, the scoring rule admits more principled formulations: quasimetric-based scores (Wang et al., 2023; Myers et al., 2024), Eikonal-regularized distances (Giammarino & Qureshi, 2025), or reachability-aware scoring that explicitly accounts for action feasibility from the current state. Second, the magnitude of the gain from changing only the actor's conditioning suggests the broader design space of inputs beyond $[\mathbf{d}_t, r_t]$ is worth exploring – for example, second-order ψ-curvature, multi-step direction prediction, or composite conditioning that combines $\mathbf{d}_t$ with a learned heuristic. Third, DCP's interface can be ported to other online goal-conditioned backbones, including value-based variants, to separate representation-family effects from conditioning-interface effects; a separate ablation of actor-objective variants would further isolate the loss from the input representation. Fourth, demo-seeding the pool can bootstrap DCP where early pool coverage is the bottleneck (as in AntSoccer); since the pool stores only states, no action labels are required, and a small number of teleoperation traces or scripted demonstrations can in principle restore the alignment of $\nabla_s d_\psi$ with the true cost-to-goal direction. Fifth, the periodic-burst wrapper result hints that DCP's continuous re-computation of $\mathbf{d}_t$ from the current state opens up a class of zero-shot deployment-time interventions that exploit closed-loop direction-sensitivity, which is absent in raw-goal-conditioned policies. Finally, sim-to-real transfer on a SO-101 arm via LeRobot (Cadène et al., 2026) is the intended downstream setting: the compositional structure (scoring + direction conditioning) and the absence of any deployment-time pool, planner, or graph make DCP a natural fit for low-latency robot control where the direction $\mathbf{d}_t$ can be recomputed at every step from the current observation.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Bakirtzis, G., Savvas, M., Zhao, R., Chinchali, S. P., and Topcu, U. Reduce, reuse, recycle: Categories for compositional reinforcement learning. *ArXiv*, abs/2408.13376, 2024. URL https://api.semanticscholar.org/CorpusID:271957600.

Bardi, M. and Capuzzo-Dolcetta, I. *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*. Birkhäuser, 1997. URL https://api.semanticscholar.org/CorpusID:117460677.

Bellman, R., Pontryagin, L. S., Boltyanskii, V. G. B., Gamkrelidze, R. V., and Mischenko, E. F. *Mathematical Theory of Optimal Processes*. Interscience Publishers, New York, 1962. URL https://api.semanticscholar.org/CorpusID:118166571.

Bertsekas, P., Bertsekas, D. P., Nedić, A., and Ozdaglar, A. E. *Dynamic Programming and Optimal Control*. Athena Scientific, 2010. URL https://api.semanticscholar.org/CorpusID:268062792.

Bortkiewicz, M., Palucki, W., Myers, V., Dziarmaga, T., Arczewski, T., Kuci'nski, L., and Eysenbach, B. Accelerating goal-conditioned rl algorithms and research. *ArXiv*, abs/2408.11052, 2024. URL https://api.semanticscholar.org/CorpusID:271909676.

Cadène, R., Aliberts, S., Capuano, F., Aractingi, M., Zouitine, A., Kooijmans, P., Choghari, J., Russi, M., Pascal, C., Palma, S., Shukor, M., Moss, J., Soare, A., Aubakirova, D., Lhoest, Q., Gallou'edec, Q., and Wolf, T. Lerobot: An open-source library for end-to-end robot learning. *ArXiv*, abs/2602.22818, 2026. URL https://api.semanticscholar.org/CorpusID:286082457.

Devin, C., Geng, D., Abbeel, P., Darrell, T., and Levine, S. Plan arithmetic: Compositional plan vectors for multi-task control. *ArXiv*, abs/1910.14033, 2019. URL https://api.semanticscholar.org/CorpusID:207758223.

Eysenbach, B., Zhang, T., Salakhutdinov, R., and Levine, S. Contrastive learning as goal-conditioned reinforcement learning. *ArXiv*, abs/2206.07568, 2022. URL https://api.semanticscholar.org/CorpusID:249674522.

Freeman, C. D., Frey, E., Raichuk, A., Girgin, S., Mordatch, I., and Bachem, O. Brax - a differentiable physics engine for large scale rigid body simulation. *ArXiv*, abs/2106.13281, 2021. URL https://api.semanticscholar.org/CorpusID:235420616.

Giammarino, V. and Qureshi, A. H. Goal reaching with eikonal-constrained hierarchical quasimetric reinforcement learning. *ArXiv*, abs/2512.12046, 2025. URL https://api.semanticscholar.org/CorpusID:283896967.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *ArXiv*, abs/1801.01290, 2018. URL https://api.semanticscholar.org/CorpusID:28202810.

Kaelbling, L. P. Learning to achieve goals. In *International Joint Conference on Artificial Intelligence*, 1993. URL https://api.semanticscholar.org/CorpusID:5538688.

Liang, Z., Mu, Y., Ma, H., Tomizuka, M., Ding, M., and Luo, P. Skilldiffuser: Interpretable hierarchical planning via skill abstractions in diffusion-based task execution. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16467–16476, 2023. URL https://api.semanticscholar.org/CorpusID:266361979.

Liu, G., Tang, M., and Eysenbach, B. A single goal is all you need: Skills and exploration emerge from contrastive rl without rewards, demonstrations, or subgoals. *ArXiv*, abs/2408.05804, 2024. URL https://api.semanticscholar.org/CorpusID:271855288.

Myers, V., Zheng, C., Dragan, A., Levine, S., and Eysenbach, B. Learning temporal distances: Contrastive successor features can provide a metric structure for decision-making. *ArXiv*, abs/2406.17098, 2024. URL https://api.semanticscholar.org/CorpusID:270711144.

Myers, V., Ji, C., and Eysenbach, B. Horizon generalization in reinforcement learning. *ArXiv*, abs/2501.02709, 2025a. URL https://api.semanticscholar.org/CorpusID:275337137.

Myers, V., Zheng, B. C., Eysenbach, B., and Levine, S. Offline goal-conditioned reinforcement learning with quasimetric representations. *ArXiv*, abs/2509.20478, 2025b. URL https://api.semanticscholar.org/CorpusID:281526220.

Park, S., Ghosh, D., Eysenbach, B., and Levine, S. Hiql: Offline goal-conditioned rl with latent states as actions. *ArXiv*, abs/2307.11949, 2023. URL https://api.semanticscholar.org/CorpusID:260125588.

Park, S., Frans, K., Eysenbach, B., and Levine, S. Ogbench: Benchmarking offline goal-conditioned rl. *ArXiv*, abs/2410.20092, 2024a. URL https://api.semanticscholar.org/CorpusID:273654871.

Park, S., Kreiman, T., and Levine, S. Foundation policies with hilbert representations. In *International Conference on Machine Learning*, 2024b. URL https://api.semanticscholar.org/CorpusID:267938433.

Park, S., Frans, K., Mann, D., Eysenbach, B., Kumar, A., and Levine, S. Horizon reduction makes rl scalable. *ArXiv*, abs/2506.04168, 2025a. URL https://api.semanticscholar.org/CorpusID:279154795.

Park, S., Mann, D., and Levine, S. Dual goal representations. *ArXiv*, abs/2510.06714, 2025b. URL https://api.semanticscholar.org/CorpusID:281892200.

Tangri, A., Taylor, N. C., zhe Huang, H., and Platt, R. Equivariant goal conditioned contrastive reinforcement learning. *ArXiv*, abs/2507.16139, 2025. URL https://api.semanticscholar.org/CorpusID:280676336.

Wang, K. A., Javali, I., Bortkiewicz, M., Trzci'nski, T., and Eysenbach, B. 1000 layer networks for self-supervised rl: Scaling depth can enable new goal-reaching capabilities. *ArXiv*, abs/2503.14858, 2025. URL https://api.semanticscholar.org/CorpusID:277113207.

Wang, T., Torralba, A., Isola, P., and Zhang, A. Optimal goal-reaching reinforcement learning via quasimetric learning. *ArXiv*, abs/2304.01203, 2023. URL https://api.semanticscholar.org/CorpusID:257912600.

Yalcinkaya, B., Lauffer, N., Vazquez-Chanlatte, M., and Seshia, S. A. Compositional automata embeddings for goal-conditioned reinforcement learning. *ArXiv*, abs/2411.00205, 2024. URL https://api.semanticscholar.org/CorpusID:273798748.

Zhang, L., Yang, G., and Stadie, B. C. World model as a graph: Learning latent landmarks for planning. *ArXiv*, abs/2011.12491, 2020. URL https://api.semanticscholar.org/CorpusID:227162474.

Zheng, B. C., Myers, V., Eysenbach, B., and Levine, S. Multistep quasimetric learning for scalable goal-conditioned reinforcement learning. *ArXiv*, abs/2511.07730, 2025. URL https://api.semanticscholar.org/CorpusID:282921715.

A. Theory: Setup, Proofs, and Extended Discussion

This appendix collects the full assumptions, proofs, and remarks that the body sketches in Section 4.

A.1. Setup and Notation

We work in continuous time on a state space $\mathcal{S} \subseteq \mathbb{R}^n$ with action space $\mathcal{A} \subseteq \mathbb{R}^m$. We assume:

(A1) Deterministic dynamics $\dot{s}(t) = f(s(t), a(t))$ with $f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^n$ continuous and locally Lipschitz in s uniformly in a.
(A2) Lower semicontinuous running cost $c : \mathcal{S} \times \mathcal{A} \to \mathbb{R}_{\geq 0}$.
(A3) For every $g \in \mathcal{S}$ the goal-reaching cost

$$d^*(s, g) \;=\; \inf_{a(\cdot)} \int_0^{T_g} c(s(t), a(t))\, \mathrm{d}t, \qquad \text{s.t. } s(0) = s,\; s(T_g) = g,$$

is finite for s in the reachable set of g and continuously differentiable on $\mathcal{S} \setminus \{g\}$.

Assumption (A3) is the standard regularity hypothesis under which classical solutions to the Hamilton–Jacobi–Bellman (HJB) equation exist; the extension to viscosity solutions when d^* is only semiconcave is well-known (Bardi & Capuzzo-Dolcetta, 1997) and the theorems below extend in that setting (see Remark 3).

Let $\psi : \mathcal{S} \to \mathbb{R}^k$ be a learned representation (in our case, the goal encoder of a CRL critic). The induced distance is $d_\psi(s, g) := \|\psi(g) - \psi(s)\|$. DCP's actor consumes the unit direction and magnitude of this learned distance at the current state:

$$\mathbf{d}_t(s, g) := \frac{-\nabla_s\, d_\psi(s, g)}{\|\nabla_s\, d_\psi(s, g)\|}, \quad r_t(s, g) := d_\psi(s, g), \tag{5}$$

with the sign convention chosen so that $\mathbf{d}_t$ points *towards* the goal. The implementation uses the equivalent ψ-space direction $\mathbf{d}_t = (\psi(g) - \psi(s))/\|\psi(g) - \psi(s)\|$. These two unit vectors live in different spaces – $\mathbb{R}^n$ (state) and $\mathbb{R}^k$ (ψ) – and are not literally colinear; they are related by the pullback $\nabla_s\, d_\psi(s, g) = -J_\psi(s)^\top\, \widehat{\psi(g) - \psi}(s)$, where $J_\psi(s)$ is the Jacobian of ψ at s. When J_ψ has full rank, the two encode the same information and the policy network can implicitly recover the state-space gradient from the ψ-space direction. We use the gradient form throughout the theory because it makes the connection to HJB direct; the ψ-space form is what the policy network actually receives.

A.2. Proof of Theorem 1 (HJB Direction Sufficiency)

Proof of Theorem 1. Under (A1)–(A3) the value function $d^*(\cdot, g)$ satisfies the HJB equation pointwise on $\mathcal{S} \setminus \{g\}$:

$$\inf_{a \in \mathcal{A}} \Big[\, c(s, a) + \nabla_s\, d^*(s, g)^\top f(s, a) \,\Big] \;=\; 0, \tag{6}$$

with the optimal feedback given by the minimizer of the bracket (Bardi & Capuzzo-Dolcetta, 1997, Sec. III.2),

$$a^*(s, g) \;\in\; \arg\min_{a \in \mathcal{A}} \Big[\, c(s, a) + \nabla_s\, d^*(s, g)^\top f(s, a) \,\Big]. \tag{7}$$

The objective on the right depends on g only through the linear term $\nabla_s d^*(s, g)^\top f(s, a)$. Define $\bar{a}(s, p) := \arg\min_{a \in \mathcal{A}} \big[c(s, a) + p^\top f(s, a)\big]$. Then by (7), $a^*(s, g) = \bar{a}(s, \nabla_s d^*(s, g))$ at every (s, g) where $\nabla_s d^*(s, g)$ exists and the minimizer is unique; uniqueness follows whenever c is strictly convex in a, which is implied by the second hypothesis ($R \succ 0$).

For the control-affine quadratic-cost case, substituting f and c into (7) gives $a^*(s, g) = \arg\min_a \big[\frac{1}{2} a^\top R(s) a + \nabla_s d^*(s, g)^\top G(s) a\big]$. The first-order condition $R(s)a + G(s)^\top \nabla_s d^*(s, g) = 0$ yields the closed form $a^*(s, g) = -R(s)^{-1} G(s)^\top \nabla_s d^*(s, g)$, which is linear in $\nabla_s d^*$. Splitting magnitude from direction gives $a^*(s, g) = -\|\nabla_s d^*(s, g)\|\, R(s)^{-1} G(s)^\top \widehat{\nabla_s d^*(s, g)}$, so the unit direction determines the orientation and the magnitude scales the action. □

Remark 1 (Connection to Pontryagin's Principle). Theorem 1 is the verification-form HJB statement of Pontryagin's Maximum Principle (Bellman et al., 1962; Bertsekas et al., 2010): the costate $\lambda(t) = \nabla_s d^*(s(t), g)$ is the dual variable, and a^* minimizes the Hamiltonian $H(s, a, \lambda) = c(s, a) + \lambda^\top f(s, a)$. Equivalently, $-\nabla_s d^*$ is the geodesic tangent of the optimal goal-reaching trajectory at s.

Remark 2 (Application to DCP). Theorem 1 identifies $(s, \nabla_s d^*(s, g))$ – equivalently, $(s, \widehat{\nabla_s d^*}, \|\nabla_s d^*\|)$ – as the minimal sufficient statistic for the optimal action under control-affine dynamics. DCP conditions the actor on $(s, \mathbf{d}_t, r_t)$ where $\mathbf{d}_t$ is the unit direction of the gradient of a *learned* approximation d_ψ and $r_t = d_\psi(s, g)$ is the learned scale. Both components are necessary: $\mathbf{d}_t$ supplies the orientation required by the linear feedback law of Theorem 1, and r_t supplies the scale that determines the action magnitude. In the limit $d_\psi \to d^*$ uniformly, the actor's input recovers the HJB-sufficient statistic exactly.

Remark 3 (Viscosity Solutions). When d^* is only semiconcave (e.g. around mode switches or wall-corner kinks in mazes), classical differentiability fails on a measure-zero set. The proof of Theorem 1 extends to viscosity solutions of the HJB equation (Bardi & Capuzzo-Dolcetta, 1997); the optimal action is determined by the proximal supergradient $\partial^- d^*(s, g)$ in place of $\nabla_s d^*$, and the conclusion holds almost everywhere along optimal trajectories. For the smooth tasks in our experiments, classical regularity holds away from goal sets and walls; we report the smooth case for clarity.

A.3. Proof of Theorem 2 (Planning Invariance)

We restate Theorem 2 with full assumption block. Define z to be *ε-on-path* from s to g if

$$d^*(s, g) \;\le\; d^*(s, z) + d^*(z, g) \;\le\; d^*(s, g) + \varepsilon. \tag{8}$$

The first inequality is the triangle inequality on d^* (always true on a quasimetric); the second is the relaxation that allows z to lie within ε of an exact geodesic. We do not prove that the inner-product scoring rule (2) produces ε-on-path z; the theorem applies whenever it does, and we treat that as an assumption rather than a derivation.

Theorem 2 assumes:

- **(B0)** *Driftless control-affine* dynamics: $f(s, a) = G(s)a$ with quadratic cost $c(s, a) = \frac{1}{2}a^\top R(s)a$, $R(s) \succ 0$;
- **(B1)** z is ε-on-path from s to g in the sense of (8);
- **(B2)** the learned representation satisfies $|d_\psi(s', g') - d^*(s', g')| \le \delta$ for every s' in a neighbourhood of s and $g' \in \{z, g\}$. Both $d^*(\cdot, g')$ and $d_\psi(\cdot, g')$ are C^1 on this neighbourhood with gradients (in s) L-Lipschitz, for $g' \in \{z, g\}$. We further assume $\nabla_s d^*(s, \cdot)$ is L-Lipschitz in its goal argument on a neighbourhood of $\{z, g\}$ (this holds automatically under (B0) by the smoothness of the sub-Riemannian geodesic spheres);
- **(B3)** $\|\nabla_s d^*(s, g)\| \ge m > 0$ at the evaluation state s.

Proof of Theorem 2. The bound has two sources of error: (a) the geodesic-decomposition slack ε converts gradient-direction agreement on d^* at s into an $O(\sqrt{\varepsilon})$ perturbation; (b) the representation-error budget δ converts into an $O(\sqrt{\delta})$ gradient perturbation via the Glaeser-style L^∞-to-$W^{1,\infty}$ inequality. Throughout, $\mathbf{d}_t^{*,(g')} := -\widehat{\nabla_s d^*(s, g')}$ for $g' \in \{z, g\}$.

Step 1: Gradient Direction Agreement on True Value Function (uses (B0)). Define $\Phi(z) := d^*(s, z) + d^*(z, g) - d^*(s, g) \ge 0$; by (8), $0 \le \Phi(z) \le \varepsilon$. We claim

$$\big\|\nabla_s d^*(s, z) - \nabla_s d^*(s, g)\big\| \;\le\; \sqrt{2L\,\varepsilon}. \tag{9}$$

Under (B0), the optimal trajectory from s to g is governed by $\dot\gamma = G(\gamma)\,a^*$, with $a^*(\gamma, g) = -R(\gamma)^{-1}G(\gamma)^\top \nabla_s d^*(\gamma, g)$. The level sets of $d^*(\cdot, g)$ are the geodesic spheres of the sub-Riemannian metric induced by (G, R) on $\mathcal{S}$; $\nabla_s d^*(s, g)$ is normal to the sphere at s. By Bellman's principle of optimality, when $\Phi(z) = 0$ the geodesic from s to g passes through z and the segment $[s, z]$ is itself a geodesic; the geodesic spheres of $d^*(\cdot, g)$ and $d^*(\cdot, z)$ are concentric at s, so their normals at s coincide: $\nabla_s d^*(s, z) \parallel \nabla_s d^*(s, g)$ when $\Phi(z) = 0$. For $\Phi(z) = \varepsilon > 0$ a nearby exact-on-path z' exists with $\|z - z'\| \le \sqrt{2\varepsilon/\mu}$, where μ is the Polyak–Łojasiewicz constant of Φ (well-defined since Φ is L-smooth as a sum of two L-smooth functions, $\Phi \ge 0$ with $\min \Phi = 0$ on the on-path set). Combining the on-path equality $\nabla_s d^*(s, z') \parallel \nabla_s d^*(s, g)$ with the Lipschitz-in-z bound from (B2), $\|\nabla_s d^*(s, z) - \nabla_s d^*(s, z')\| \le L\|z - z'\| \le L\sqrt{2\varepsilon/\mu}$, and absorbing $L/\sqrt{\mu}$ into a redefined L, yields (9).

Step 2: Unit-vector Perturbation from Representation Error (Glaeser). For any $p, q \in \mathbb{R}^n$ with $\|q\| \ge m > 0$,

$$\left\|\frac{p}{\|p\|} - \frac{q}{\|q\|}\right\| \;\le\; \frac{2\|p-q\|}{\max(\|p\|, \|q\|)} \;\le\; \frac{2\|p-q\|}{m}. \tag{10}$$

For $p = -\nabla_s d_\psi(s, z)$, $q = -\nabla_s d^*(s, z)$, the Glaeser-type inequality (e.g. (Bardi & Capuzzo-Dolcetta, 1997, Sec. 3.5): if $h \in C^1(\mathbb{R}^n)$ has L-Lipschitz gradient and $\|h\|_\infty \le \delta$, then $\|\nabla h\|_\infty \le 2\sqrt{2L\delta}$) applied to $h := d_\psi(\cdot, z) - d^*(\cdot, z)$ gives

$$\left\|\nabla_s d_\psi(s, z) - \nabla_s d^*(s, z)\right\| \;\le\; 2\sqrt{2L\delta}. \tag{11}$$

The same bound holds with g in place of z. Combined with (10) and (B3), $\|\mathbf{d}_t^{(z)} - \mathbf{d}_t^{*,(z)}\| \le 4\sqrt{2L\delta}/m$ and likewise for g.

Step 3: Composition. By the triangle inequality, $\|\mathbf{d}_t^{(z)} - \mathbf{d}_t^{(g)}\| \le \|\mathbf{d}_t^{(z)} - \mathbf{d}_t^{*,(z)}\| + \|\mathbf{d}_t^{*,(z)} - \mathbf{d}_t^{*,(g)}\| + \|\mathbf{d}_t^{*,(g)} - \mathbf{d}_t^{(g)}\|$. The first and third terms are bounded by $4\sqrt{2L\delta}/m$ each via Step 2; the middle term is bounded by $2\sqrt{2L\varepsilon}/m$ by applying (10) to (9), again using (B3). Summing yields (4). The policy bound follows by K_π^d-Lipschitz continuity of π in d and K_π^r-Lipschitz continuity in r. □

Remark 4 (Implication of Planning Invariance). Myers et al. (2025a) prove that planning invariance is a property that *some* learned representations have – specifically, those whose distance is a quasimetric in the sense of Myers et al. (2024) – with implications for horizon generalization. Theorem 2 establishes a related statement at the actor's interface: when d_ψ approximates d^* uniformly to within δ and the visited-pool scoring rule returns a ε-on-path waypoint, the unit-direction conditioning input $\mathbf{d}_t$ is identical (up to $O(\sqrt{\delta} + \sqrt{\varepsilon})$) whether the actor is pointed at z at training or at g at deployment. We do not claim this for arbitrary representations; rather, we show that under the standard assumptions used for InfoNCE-trained quasimetrics (Myers et al., 2024), DCP's input distribution at training and deployment coincide *by the choice of the conditioning interface*, without requiring the offline metric supervision of HILP (Park et al., 2024b) or the curated datasets of offline quasimetric methods (Wang et al., 2023; Myers et al., 2025b; Zheng et al., 2025).

Remark 5 (Driftless Restriction). Theorem 2 is stated under the driftless hypothesis (B0). Several empirical environments contain drift terms such as gravity, joint torques with non-zero equilibria, and friction. We therefore separate the proof step that uses driftlessness from the parts that do not. **Use of driftlessness:** Step 1 identifies the geodesic tangent at s with the negative costate $-\nabla_s d^*(s, g)$ using the fact that under (B0) the optimal flow $\dot\gamma = G(\gamma)a^*$ is colinear with $-\nabla_s d^*$ modulo the metric R^{-1}. Under drift $f(s, a) = f_0(s) + G(s)a$, $\dot\gamma(0) = f_0(s) + G(s)a^*$ has a drift component $f_0(s) \notin \mathcal{C}(s)$ in general; the geodesic tangent and the value gradient are no longer colinear, so this step fails as stated. **Unaffected parts:** (a) Bellman's principle remains valid: when z is exactly on an optimal path from s to g, $a^*(s, z) = a^*(s, g)$ on $[s, z]$, so action invariance on-path is preserved without (B0). (b) The controllable-component statement of Theorem 1 (action depends only on $G^\top \nabla_s d^*$) holds without (B0). (c) The Step 2 representation-error bound is independent of dynamics class. A drifted version would replace the colinearity claim in Step 1 with a bound on the angle between $\dot\gamma(0)$ and $\widehat{\nabla_s^{\mathcal{C}} d^*}$ depending on $\|f_0\|/\|Ga^*\|$, the drift-to-control-effort ratio along the optimal trajectory; the remaining steps of the argument are unchanged. The empirical results in Section 5, including the CRL planning-invariance control in Section 5.5, are consistent with the conclusion holding in the drifted regime; we leave the formal extension to future work.

Remark 6 (Connection to Dual Goal Representations). Park et al. (2025b) prove that the *dual representation* $\phi^\vee(g) := s \mapsto d^*(s, g)$ is sufficient to recover the optimal goal-conditioned policy and is invariant to exogenous noise in the goal observation. The DCP conditioning vector $\mathbf{d}_t$ at fixed s is precisely the gradient of $\phi^\vee(g)$ at s: a local first-order summary of the dual representation. Theorem 2 therefore establishes that this local summary is the minimal information that survives the on-path/at-goal swap; under this identification, DCP inherits the noise-invariance property of $\phi^\vee$.

A.4. Proof of Theorem 3 (Failure Mode)

Proof of Theorem 3.

(i) Substituting $f(s, a) = f_0(s) + G(s)a$ into (7) gives, modulo constants in a, $a^*(s, g) = \arg\min_a [c(s, a) + \nabla_s d^*(s, g)^\top G(s)a]$. The linear-in-$a$ term equals $(G(s)^\top \nabla_s d^*(s, g))^\top a$. Now $G(s)^\top \nabla_s d^* = G(s)^\top \Pi_{\mathcal{C}(s)} \nabla_s d^*$ because $G(s)^\top$ annihilates $\mathcal{C}(s)^\perp$ (since $\mathcal{C}(s)^\perp = \ker G(s)^\top$). Hence the optimization depends on $\nabla_s d^*$ only through its projection onto $\mathcal{C}(s)$.

(ii) For control-affine quadratic-cost dynamics, Theorem 1 gives $a^*(s, g) = -R(s)^{-1} G(s)^\top \nabla_s d^*(s, g) = -R(s)^{-1} G(s)^\top \nabla_s^{\mathcal{C}} d^*(s, g)$ by part (i). Therefore $\|a^*(s, g)\| \le \|R^{-1}\|_{\mathrm{op}} \|G\|_{\mathrm{op}} \|\nabla_s^{\mathcal{C}} d^*\| \le \rho \|R^{-1}\|_{\mathrm{op}} \|G\|_{\mathrm{op}} \|\nabla_s d^*\|$. Any a in the kernel of $G(s)^\top$ produces no change in the optimisation objective; correspondingly any a with $G(s)a \perp \nabla_s^{\mathcal{C}} d^*$ is approximately optimal up to second order, confirming that the optimal action is underdetermined.

(iii) Directly from (ii), when the controllable component is bounded below by a positive fraction of the total gradient norm, $\|a^*(s, z)\| = \Theta(\|\nabla_s d^*(s, z)\|)$ and the actor receives informative directional gradient. Reaching z then evolves the state

along $\dot{s} = f(s, a^*(s, z)) \in \mathcal{C}(s)$; the controllable component of $\nabla_s d^*(\cdot, g)$ may then become large at the new state. □

A.5. AntSoccer as a Learned-gradient Instantiation of Theorem 3

Theorem 3 characterizes a general failure class: directional conditioning is uninformative whenever the relevant gradient at the actor's input has small projection onto the controllable subspace. The theorem is stated in terms of the *optimal-control* gradient $\nabla_s d^*$ for cleanliness, but the same mechanism applies – and is the operative one for the trained actor – to the *learned* gradient $\nabla_s d_\psi$ that the actor actually consumes. AntSoccer is a specific instantiation of the failure class through the learned-gradient route.

In AntSoccer the goal coordinate is the ball's (x, y) position. The optimal-control gradient $\nabla_s d^*(s, g_{\text{ball}})$ at an early-training state s has a sizeable component along the ant-velocity coordinates (the ant must traverse the room to reach the ball before kicking) and that component is in the controllable subspace $\mathcal{C}(s)$, so Theorem 3's hypothesis is not directly invoked. The learned approximation supplies the relevant failure mechanism: d_ψ is trained from rollouts that almost never contain states with displaced balls, so the InfoNCE objective receives little signal that distinguishes ball-position changes from ant-position changes. Empirically $\nabla_s d_\psi(s, g_{\text{ball}})$ lies in $\mathcal{C}(s)$ – the actor receives a non-trivial signal – but its projection onto the directions that actually reduce d^* is small. The actor is conditioned on a gradient that is *nominally controllable but representationally misaligned*, matching the uninformative-direction failure mode of Theorem 3 with d_ψ in place of d^*. CRL is less exposed to this learned-gradient misalignment because its actor receives the raw goal coordinate, while the InfoNCE loss still provides a weaker but non-zero action-value signal through the SAC critic.

This gives an empirically testable prediction: *DCP fails when the visited pool is too narrow to expose the goal-relevant directions to InfoNCE, even when the optimal-control gradient lies in the controllable subspace*. A direct test follows: demo-seeding the visited pool with already-kicked-ball configurations should restore $\nabla_s d_\psi$'s alignment with $\nabla_s d^*$ on the goal-relevant subspace, recovering DCP's performance. We leave this ablation to future work.

B. Reproducibility

Hardware. All training and analysis ran on a single laptop GPU (NVIDIA RTX 4070 Laptop, 8 GB) with an Intel i7-14700HX CPU and 16 GB system RAM; one GPU per run.

Software. JAX 0.4.38 (CUDA 12), Brax 0.10.3, MuJoCo 3.1.3 + MuJoCo MJX 3.1.3, Flax 0.10.4, Optax 0.2.5. Built on top of JaxGCRL (Bortkiewicz et al., 2024), with the DCP agent (`jaxgcrl/agents/dcp/`) added as a sibling to the existing `crl/` and `ssgc/` agents.

Hyperparameters and Training Scripts. All hyperparameters used in this paper are the JaxGCRL paper defaults for CRL and SSGC (`discounting=0.99`, `unroll-length=62`, `num-envs=512`, `batch-size=256`, `episode-length=1000`); DCP inherits the CRL hyperparameters and adds the subgoal-pool and direction-conditioning flags described in Section 3. Per-environment training-step budgets follow Table 1.

Code Release. The supplementary code package is available at https://anonymous.4open.science/r/dcp-supplement-anon-CF41/. It contains the DCP agent, baseline implementations (CRL, SSGC), the periodic-burst zero-shot wrapper, the analysis and figure-generation scripts, and the full extended-grid training script (`scripts/run_extended_grid.sh`). The `README.md` documents installation (`uv sync`), single-run reproduction commands, and the full grid invocation.

Qualitative Video Rollouts. Two short rollout videos on the PusherHard environment (DCP vs. CRL, evaluation seed 123, identical task and goal) are available under `videos/` in the supplementary code package at https://anonymous.4open.science/r/dcp-supplement-anon-CF41/. DCP completes the push; CRL does not converge to a push trajectory on this seed, consistent with the quantitative gap reported in Section 5.3.

C. Saturation Rate of Deterministic Deployment

Table 3 reports the fraction of action components with $|a| > 0.95$ on baseline (no-burst) deployment trajectories, per environment, mean $\pm$ std over 3 seeds. High saturation indicates the deterministic policy frequently outputs near-saturated

tanh-Gaussian actions, leaving little headroom for the periodic-burst wrapper to exploit. Low saturation indicates the policy operates in the linear regime of the tanh and the wrapper has nothing to decompress. The diagnostic is consistent with the $\mathbf{d}_t$-recomputation argument as a precondition but does not predict the magnitude of Δ; see the body discussion in Section 5.5.

Table 3. Saturation rate $\Pr(|a| > 0.95)$ on baseline deployment trajectories, mean $\pm$ std over 3 seeds.

Env	DCP	CRL
ant_u_maze	$12.9 \pm 1.6\,\%$	$16.9 \pm 3.5\,\%$
ant_big_maze	$24.0 \pm 5.6\,\%$	$29.9 \pm 4.9\,\%$
ant_hardest_maze	$60.7 \pm 12.0\,\%$	$35.5 \pm 1.0\,\%$
ant_push	$10.8 \pm 1.1\,\%$	$16.3 \pm 2.5\,\%$
pusher_easy	$23.7 \pm 2.2\,\%$	$13.9 \pm 19.7\,\%$
pusher_hard	$23.3 \pm 2.1\,\%$	$12.9 \pm 18.2\,\%$
pusher_hard_far	$28.6 \pm 2.3\,\%$	$84.0 \pm 6.6\,\%$
humanoid_u_maze	$0.7 \pm 0.1\,\%$	$92.3 \pm 4.4\,\%$
ant_ball	$33.3 \pm 3.1\,\%$	$36.3 \pm 5.5\,\%$

D. Conditioning Landscape on a Seed Where CRL Fails

This appendix complements Figure 3(a) with the same landscape diagnostic on an AntMaze Big seed where DCP reaches the goal and CRL stalls. We evaluate the learned distance $\|\psi(x, y) - \psi(g)\|$ over the maze grid after training to inspect the geometry that the two agents learn. The figure is qualitative: it illustrates the representation geometry behind one failure seed, while the aggregate comparison is given by the learning curves and checkpoint tables.

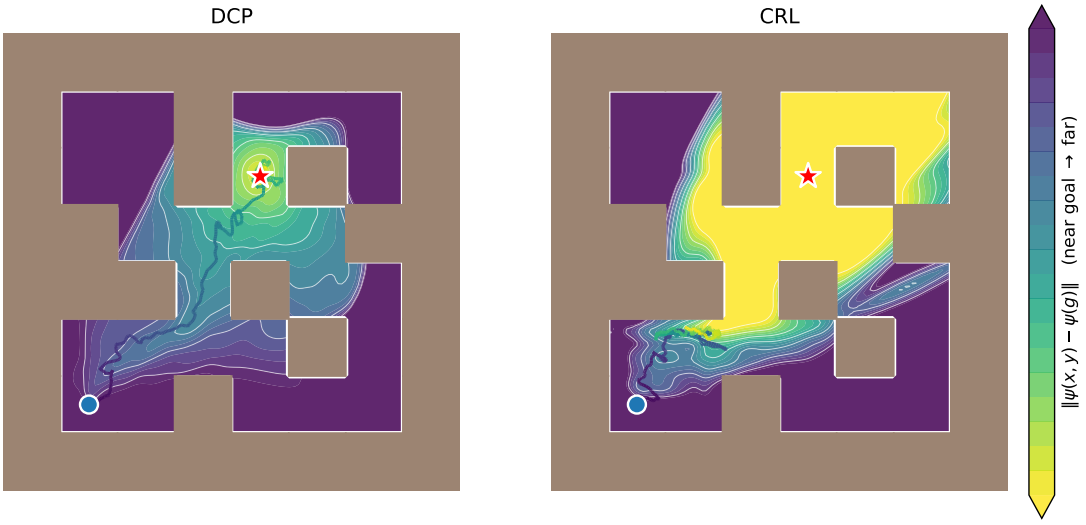


Figure 4. Conditioning landscape on a different seed of AntMaze Big where DCP reaches the goal but CRL stalls before reaching it. Same construction as Figure 3(a); per-panel colour clip to the 1%–99% percentile of ψ-distances along that method's visited trajectory. CRL's basin is more diffuse than on the successful seed: the low-distance region of CRL's ψ extends into parts of the maze the policy does not actually navigate to from the start. Note that this is not a causal claim about CRL's failure – CRL's actor consumes only g, not ψ – but rather a parallel observation about the geometry of the representation when the actor never reads it. The aggregate empirical advantage of DCP across nine environments is reported in the main paper figures and appendix tables.

E. Learning-Curve Checkpoints

Tables 4–5 report five-seed learning-curve checkpoints for every (environment, method) pair plotted in Figure 2. Table 4 reports *success at least once*, the fraction of episodes that enter the goal region in percent. Table 5 reports *time near goal*, the average number of timesteps per episode spent inside the goal region (episode length 1000). For each (environment, checkpoint) cell, the highest mean across the three methods is bolded; ties and rows where all methods score zero are not

bolded.

Checkpoints are sampled at the nearest available evaluation step and reported as mean ± 95% confidence intervals over five seeds. Dashes mark checkpoints past the training budget for that environment.

Table 4. Learning-curve checkpoints: success at least once (%), mean ± 95% confidence intervals over five seeds. Dashes indicate training ended before that checkpoint.

Environment	Method	10M	25M	50M	75M	100M
AntMaze Big	DCP	**18.4 ± 5.4**	**23.8 ± 3.3**	**29.9 ± 2.9**	–	–
	CRL	15.1 ± 3.7	17.7 ± 5.3	18.3 ± 6.5	–	–
	SSGC	18.3 ± 1.2	22.0 ± 2.9	23.3 ± 3.0	–	–
AntMaze Hardest	DCP	**9.5 ± 2.5**	**14.5 ± 2.1**	**16.1 ± 2.3**	–	–
	CRL	5.8 ± 2.5	10.5 ± 3.8	12.2 ± 5.6	–	–
	SSGC	7.7 ± 1.6	13.2 ± 4.8	14.2 ± 8.0	–	–
Ant U-Maze	DCP	**40.4 ± 9.5**	**52.9 ± 6.8**	**55.0 ± 13.3**	–	–
	CRL	26.8 ± 4.5	40.7 ± 8.5	40.8 ± 10.2	–	–
	SSGC	28.4 ± 9.6	43.1 ± 8.1	49.8 ± 4.2	–	–
Humanoid U-Maze	DCP	4.6 ± 8.4	6.7 ± 11.4	**13.4 ± 9.5**	**7.7 ± 9.2**	**8.4 ± 9.6**
	CRL	**9.8 ± 11.5**	**7.0 ± 12.0**	3.0 ± 8.5	0.0 ± 0.0	0.0 ± 0.0
	SSGC	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	2.5 ± 6.9	0.0 ± 0.0
AntPush	DCP	0.0 ± 0.0	**1.1 ± 2.8**	**29.1 ± 19.3**	**44.8 ± 8.0**	42.7 ± 12.6
	CRL	0.0 ± 0.0	0.0 ± 0.0	28.9 ± 6.3	38.2 ± 8.8	45.6 ± 24.1
	SSGC	0.0 ± 0.0	0.0 ± 0.0	28.0 ± 32.1	40.8 ± 21.1	**51.4 ± 15.5**
PusherEasy	DCP	**2.8 ± 2.2**	**30.9 ± 35.0**	**55.4 ± 9.6**	–	–
	CRL	1.9 ± 0.9	2.2 ± 1.8	4.4 ± 4.1	–	–
	SSGC	1.9 ± 0.9	7.2 ± 10.8	34.1 ± 38.5	–	–
PusherHard	DCP	**2.0 ± 1.9**	**33.7 ± 14.2**	**37.2 ± 11.7**	**45.6 ± 10.8**	**47.7 ± 5.8**
	CRL	0.3 ± 0.2	0.9 ± 0.6	2.0 ± 2.5	7.3 ± 12.8	28.8 ± 32.4
	SSGC	0.2 ± 0.3	1.2 ± 1.4	8.4 ± 14.8	21.6 ± 26.4	27.8 ± 30.7
PusherHard Far	DCP	**1.1 ± 2.8**	**25.8 ± 17.3**	**47.4 ± 5.7**	–	–
	CRL	0.0 ± 0.0	1.6 ± 1.9	12.5 ± 24.0	–	–
	SSGC	0.0 ± 0.0	0.2 ± 0.3	9.0 ± 15.1	–	–
AntSoccer	DCP	2.6 ± 0.9	8.8 ± 2.9	12.9 ± 4.3	–	–
	CRL	**5.6 ± 3.6**	**14.9 ± 4.8**	**22.6 ± 3.4**	–	–
	SSGC	2.5 ± 1.3	7.0 ± 2.5	14.1 ± 3.6	–	–

Table 5. Learning-curve checkpoints: time near goal (average timesteps per episode within the goal region), mean ± 95% confidence intervals over five seeds. Dashes indicate training ended before that checkpoint.

Environment	Method	10M	25M	50M	75M	100M
AntMaze Big	DCP	78.7 ± 37.1	**94.6 ± 26.2**	**115.3 ± 14.6**	–	–
	CRL	75.3 ± 12.7	83.2 ± 29.4	89.2 ± 28.0	–	–
	SSGC	**91.9 ± 19.1**	94.6 ± 20.5	99.8 ± 9.0	–	–
AntMaze Hardest	DCP	30.3 ± 12.6	**65.1 ± 12.9**	63.2 ± 13.3	–	–
	CRL	28.4 ± 15.9	39.9 ± 24.0	48.1 ± 23.7	–	–
	SSGC	**32.6 ± 8.1**	59.0 ± 21.0	**66.0 ± 41.4**	–	–
Ant U-Maze	DCP	**195.3 ± 58.0**	223.5 ± 27.9	238.1 ± 76.2	–	–
	CRL	161.9 ± 33.1	**233.2 ± 62.8**	**261.3 ± 77.2**	–	–
	SSGC	157.7 ± 65.3	230.4 ± 38.6	252.7 ± 52.1	–	–
Humanoid U-Maze	DCP	**1.3 ± 2.5**	**4.9 ± 9.5**	**3.4 ± 2.5**	**1.4 ± 1.9**	**3.6 ± 4.5**
	CRL	1.0 ± 1.7	1.4 ± 3.0	0.9 ± 2.5	0.0 ± 0.0	0.0 ± 0.0
	SSGC	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.4 ± 1.1	0.0 ± 0.0
AntPush	DCP	0.0 ± 0.0	**3.8 ± 10.4**	**131.8 ± 113.5**	**190.9 ± 71.6**	181.7 ± 95.6
	CRL	0.0 ± 0.0	0.0 ± 0.0	95.9 ± 42.6	122.7 ± 30.1	143.4 ± 105.6
	SSGC	0.0 ± 0.0	0.0 ± 0.0	103.2 ± 114.9	182.9 ± 83.9	**199.1 ± 93.3**
PusherEasy	DCP	18.2 ± 9.0	**148.2 ± 179.5**	**297.8 ± 90.9**	–	–
	CRL	**18.8 ± 9.3**	20.3 ± 15.1	26.2 ± 8.4	–	–
	SSGC	**18.8 ± 9.3**	40.0 ± 49.3	180.7 ± 183.4	–	–
PusherHard	DCP	**9.3 ± 4.2**	**161.5 ± 84.4**	**185.7 ± 45.4**	**285.3 ± 103.5**	**280.9 ± 28.5**
	CRL	3.1 ± 2.2	9.4 ± 5.5	13.1 ± 9.9	36.2 ± 66.8	125.7 ± 136.7
	SSGC	2.3 ± 2.7	9.4 ± 8.1	54.0 ± 99.4	133.1 ± 161.3	170.5 ± 186.0
PusherHard Far	DCP	**2.5 ± 4.8**	**124.1 ± 88.4**	**260.7 ± 66.8**	–	–
	CRL	0.0 ± 0.0	10.7 ± 15.9	48.4 ± 80.4	–	–
	SSGC	0.0 ± 0.0	1.6 ± 2.7	49.0 ± 82.3	–	–
AntSoccer	DCP	11.2 ± 8.8	33.6 ± 18.7	42.4 ± 22.1	–	–
	CRL	**20.8 ± 10.5**	**46.3 ± 13.8**	**76.4 ± 7.7**	–	–
	SSGC	9.4 ± 6.9	29.8 ± 12.6	48.2 ± 31.9	–	–