FINITE-HORIZON HAMILTONIANS FOR VALUE GEOMETRY IN GOAL-CONDITIONED RL

Anonymous authors
Paper under double-blind review

ABSTRACT

In goal-conditioned reinforcement learning (GCRL), quasimetric learning models goal-reaching costs as quasimetric distances, connecting local constraints to global value geometry. Its local constraints, however, should reflect the direction-dependent effects of control composition over a finite horizon together with environmental feasibility. We propose ***ReQRL***, which constrains the critic's value gradients through finite-horizon reachability. Drawing on state-constrained optimal control, we decouple dynamical reachability from boundary geometry, estimating both from data. On `OGBench`, our method outperforms or rivals existing quasimetric approaches and other offline GCRL methods.

1 INTRODUCTION

Goal-conditioned reinforcement learning (GCRL) provides a framework for learning a shared policy across many goals, where experience collected while pursuing one goal can help the agent generalize to reach others. Specifying tasks through goals also reduces reliance on task-specific reward shaping (Schaul et al., 2015; Andrychowicz et al., 2017). A central challenge is learning optimal goal-conditioned values that describe the minimum cost of reaching each goal and guide policy learning and planning (Pong et al., 2018; Park et al., 2023). For instance, reaching a goal may require a long detour because an obstacle blocks the direct path or the available controls do not allow the agent to move directly toward it. Thus, the value function, or *critic*, must reflect this global cost of reaching a goal, accounting for all the composed actions needed to follow the optimal feasible path.

The goal-reaching costs above have a useful structure. They are nonnegative and zero when already at the goal, but may differ with the direction of travel. Moreover, the optimal cost of reaching the goal g from a state s cannot exceed the cost of going through an intermediate state s'. These properties precisely make the optimal goal-reaching costs $-V^*(s, g)$ a *quasimetric*. This distance structure has motivated various approaches (Wang et al., 2023; Myers et al., 2026; Giammarino & Qureshi, 2026), which restrict the critic's hypothesis class through quasimetric parameterization. In particular, to learn an environment's reaching costs, Quasimetric RL (QRL) (Wang et al., 2023) imposes local constraints that keep predicted distances consistent with the costs of observed feasible transitions (s, s'). On the other hand, EikHiQRL (Giammarino & Qureshi, 2026) directly constrains gradient magnitudes to regulate how rapidly its predicted distances change with state. The appeal of the quasimetric structure is that it links, or *stitches*, reaching costs through intermediate states, so local supervision can inform global goal-reaching distance estimates.

Quasimetric parameterization links reaching costs across states, but does not specify which local changes in the state are physically possible or which paths are blocked. *Local constraints* help convey this information to the critic, i.e., the learned value geometry depends on what these local constraints tell the critic about the environment's *(i) control dynamics* (how actions change the state) and *(ii) domain feasibility* (how the environment restricts feasible paths). Although QRL and EikHiQRL provide effective local supervision, their constraints remain limited. In practice, QRL enforces its local bounds on sparse sampled transitions in data. This leaves the critic to infer how its value should change along *other* feasible motions. EikHiQRL constrains local variation through a simplified isotropic dynamics model without an explicit boundary condition, which may impose a local distance geometry that does not match the agent's directional control capabilities or the domain's feasibility.

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.

This paper. We ask: *can we provide local critic supervision that faithfully reflects both the system's control dynamics and domain feasibility?* To answer this, we show this supervision need *not* rely only on instantaneous control directions but can instead retain the effects of controls composed over time. Geometric control theory (Jean, 2014) explains why: composing controls over a finite duration can produce state changes unavailable instantaneously. We capture these state changes through a *finite-horizon reachable set*, and derive a *finite-horizon Hamiltonian* encoding these effects in a gradient constraint by linearizing only the critic. To make reachability estimation tractable from transition data, we then draw on state-constrained Hamilton-Jacobi theory (Soner, 1986) to decouple dynamical reachability from boundary geometry. We thus supply two complementary local training signals: (i) what composed controls can achieve dynamically and (ii) how domain boundaries restrict feasible progress toward a goal. Together they let the critic capture global goal-reaching geometry (Figure 1).

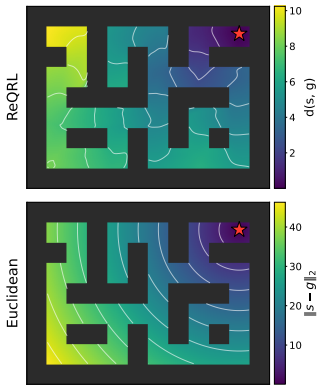

Figure 1: ReQRL's value geometry (top) and Euclidean (bottom). Colors for distance; dark regions denote obstacles, stars the goal.

Our contributions: **(a)** we formulate quasimetric local constraints through a finite-horizon Hamiltonian retaining the effects of composed controls and decouple dynamical reachability from boundary feasibility using state-constrained Hamilton-Jacobi theory, **(b)** introduce **ReQRL**, a GCRL instantiation estimated from offline transition data, with closed-form zonotopic reachability constraints and boundary regularization, and **(c)** on `OGBench`, ReQRL largely outperforms QRL and EikHiQRL in success rate, matching or exceeding recent state-of-the-art GCRL methods on many tasks.

2 Background

Goal-conditioned RL. Consider a deterministic goal-conditioned MDP $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \rho_0, \rho_g, R, \gamma = 1)$, where $\mathcal{S} \subseteq \mathbb{R}^n$ and $\mathcal{A} \subseteq \mathbb{R}^m$ are the state and action spaces. $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ is the deterministic transition map, and $R(s, g) := -\mathbb{1}\{s \neq g\}$ is the sparse goal-conditioned reward. Let ρ_0 and ρ_g denote the initial state and goal distributions. A goal-conditioned policy $\pi(\cdot \mid s, g)$ has value

$$V^\pi(s, g) := \mathbb{E}_\pi \left[\sum_{t=0}^{T-1} R(s_t, g) \,\middle|\, s_0 = s \right],$$

where T is the first goal-hitting time. The policy aims to maximize $\mathbb{E}_{s_0 \sim \rho_0, g \sim \rho_g}[V^\pi(s_0, g)]$, and the optimal value is $V^*(s, g) = \max_\pi V^\pi(s, g)$. In the offline setting, we learn from a fixed dataset $\mathcal{D} = \{\tau_i\}_{i=1}^N$ of trajectories $\tau_i = (s_0^i, a_0^i, s_1^i, \ldots, s_{T_i}^i)$ collected by an unknown behavior policy.

Quasimetric RL. A *quasimetric* is a function $d : \mathcal{S} \times \mathcal{S} \to \mathbb{R}_{\geq 0}$ with $d(s, s) = 0$ and the triangle inequality $d(s_1, s_3) \leq d(s_1, s_2) + d(s_2, s_3)$. It need not be symmetric. Let us denote by $\mathcal{Q}(\mathcal{S})$ the class of all quasimetrics on $\mathcal{S}$. QRL (Wang et al., 2023) observes that, under deterministic transitions in the undiscounted (i.e., $\gamma = 1$) unit-cost goal-reaching setting, the negative optimal value $d^*(s, g) := -V^*(s, g)$ is the directed shortest-path distance from s to g, and hence $d^* \in \mathcal{Q}(\mathcal{S})$.

QRL reframes GCRL as an optimization over $\mathcal{Q}(\mathcal{S})$. They parameterize a critic as $d_\theta \in \mathcal{Q}(\mathcal{S})$ using an Interval Quasimetric Embedding (IQE) (Wang & Isola, 2022), whose architecture guarantees $d_\theta \in \mathcal{Q}(\mathcal{S})$ for every θ, and learn it by maximizing distances between (s, g) pairs while constraining every observed transition by its local cost:

$$\max_\theta \; \mathbb{E}_{s,g}[d_\theta(s, g)] \qquad \text{s.t.} \qquad d_\theta(s, s') \leq 1, \quad \forall (s, a, s') \in \mathcal{D}. \tag{1}$$

By the triangle inequality, these local constraints bound $d_\theta(s, g)$ by the cost of any path composed of constrained transitions. When optimizing over $\mathcal{Q}(\mathcal{S})$ with sufficient transition and state–goal coverage, this objective makes the bound tight at the shortest path and recovers $d_{\theta^*} = -V^*$. The learned value can then be used for planning or policy learning, e.g., a DDPG-style actor (Lillicrap et al., 2016) or advantage-weighted regression (AWR) (Peng et al., 2019).

EikQRL (Eikonal-QRL) (Giammarino & Qureshi, 2026) derives local constraints via a continuous-time limit. They assume deterministic Lipschitz dynamics admitting $\mathcal{T}(s, a) = s + f(s, a)\Delta t +$

$o(\Delta t)$ with time step Δt and velocity $f(s,a)$, unit running cost until reaching the goal, and a finite locally Lipschitz optimal value. Taking $\Delta t \to 0$ yields the Hamilton-Jacobi-Bellman equation, which reduces under unit-speed isotropic dynamics, $f(s,a) = a$ with $\|a\|_2 \leq 1$, to the Eikonal equation $\|\nabla_s d^*\|_2 = 1$ (distance grows at unit rate). EikQRL imposes this softly with tolerance $\epsilon > 0$:

$$\max_{\theta} \mathbb{E}_{s,g}[d_\theta(s,g)] \qquad \text{s.t.} \qquad \mathbb{E}_{s,g}\left[\left(\|\nabla_s d_\theta(s,g)\|_2 - 1\right)^2\right] \leq \epsilon^2. \tag{2}$$

This gradient constraint is denser than QRL's, as it applies at all sampled (s,g) pairs rather than along observed transitions in $\mathcal{D}$. However, the isotropic reduction of dynamics assumes an agent can move at unit speed equally in every direction at every state. This also constrains the critic's local variation in the *interior* but does not explicitly enforce boundary feasibility. In effect, the constraint acts as a dynamics-agnostic regularizer on the critic gradient in the domain interior. In the implementation, EikHiQRL combines this with HIQL's temporal abstraction (Park et al., 2023).

3 Going from Control Composition to Critic Supervision

Quasimetric structure determines local information propagation through intermediate states. We ask, *what information about feasible motion should local constraints supply to the critic?* The answer depends on the time scale at which dynamics are viewed. We focus on deterministic control systems with $\mathcal{S}$ and $\mathcal{A}$.

Instantaneous controls restrict motion to a few directions, but composing controls over a nonzero duration make additional directions reachable. We begin with a simple geometric control example.

Example. Consider a simple planar motion controlled by turning and moving forward or backward with state $s = (x, y, z)$ representing its orientation angle, forward position, and sideways position. For small heading angles, a simplified model is

$$\dot{x} = a_1, \qquad \dot{y} = a_2, \qquad \dot{z} = x a_2, \qquad \|a\|_2 \leq 1,$$

where a_1 controls turning and a_2 controls motion along the current heading. Thus, the control fields are $X(s) = (1,0,0)^\top$ and $Y(s) = (0,1,x)^\top$. At the origin, every instantaneous velocity has the form $(a_1, a_2, 0) \in \mathcal{H}_0 := \text{span}\{X(0), Y(0)\}$, so no control produces instantaneous sideways motion. Let us evolve the state by following $X, Y, -X$, and $-Y$ each for duration $\Delta t > 0$. After the duration $\tau = 4\Delta t$, the system reaches

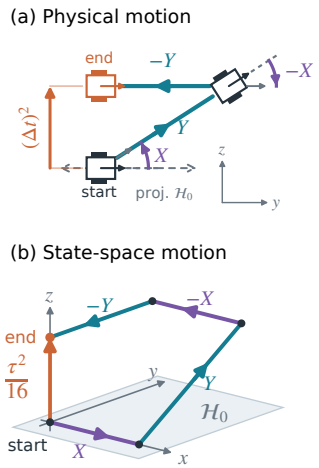

Figure 2: Composed controls produce sideways motion (a), with net state change outside the initial instantaneous span $\mathcal{H}_0$ (b).

$$(0,0,0) \xrightarrow{X} (\Delta t, 0, 0) \xrightarrow{Y} \left(\Delta t, \Delta t, (\Delta t)^2\right) \xrightarrow{-X} \left(0, \Delta t, (\Delta t)^2\right) \xrightarrow{-Y} \left(0, 0, (\Delta t)^2\right).$$

The sequence restores x and y while retaining a net change $(\Delta t)^2$ in z (Figure 2(a)). Therefore, control composition *exposes* a reachable direction which was absent from the initial instantaneous control subspace $\mathcal{H}_0$ (Figure 2(b)). This net change equals $(\Delta t)^2[X,Y](0)$, where the *Lie bracket*

$$[X,Y](s) := DY(s)X(s) - DX(s)Y(s) = (0,0,1)^\top \notin \mathcal{H}_0$$

characterizes this additional component where DX and DY are the field Jacobians (Agrachev et al., 2020). Over duration τ, the state in z evolves by $\tau^2/16$ with average velocity $(0,0,\tau/16)^\top$. Thus, the composed controls *produce sideways motion at every finite* $\tau > 0$, but the average velocity vanishes as we take the limit $\tau \to 0$.

What does this illustrate? Composing controls over a finite duration can produce motion outside the initial instantaneous control subspace (Jean, 2014). If we locally constrain the critic gradient $\nabla_s d_\theta$ only along control directions available instantaneously as $\tau \to 0$, the resulting condition stays insensitive to gradient components *orthogonal* to this subspace. However, a finite horizon $\tau > 0$ changes which components of the critic gradient a local constraint responds to. This motivates *constraining the critic d_θ explicitly with information about motions achievable through control composition.*

How does control composition shape goal-reaching values? Consider the undiscounted minimum-time problem with unit cost per unit time and a well-defined deterministic controlled

evolution. For a supported finite duration $\tau > 0$, let $\mathcal{U}_\tau(s)$ denote the *feasible* controls of duration τ from s using actions in $\mathcal{A}$. Each control determines a unique endpoint $\mathcal{T}_\tau(s, u) \in \mathcal{S}$. The standard Bellman optimality (Liberzon, 2011) gives

$$d^*(s, g) = \tau + \inf_{u \in \mathcal{U}_\tau(s)} d^*(\mathcal{T}_\tau(s, u), g), \qquad d^*(s, g) > \tau, \tag{3}$$

with $d^*(g, g) = 0$. In (3), the optimal cost is the elapsed duration τ plus the minimum remaining cost from a reachable endpoint. Notice the horizon τ determines how long controls are composed before evaluating the remaining cost. The endpoint map $u \mapsto \mathcal{T}_\tau(s, u)$ tells us which states can be reached from s by composing controls over duration τ under *both the environment's dynamics and its feasibility constraints*. These are exactly the states from which Bellman optimality evaluates the remaining goal-reaching cost, connecting composed controls to the value we aim to learn. Its image, translated by $-s$, gives the *reachable set*

$$\mathcal{R}_\tau(s) := \{\mathcal{T}_\tau(s, u) - s : u \in \mathcal{U}_\tau(s)\}. \tag{4}$$

Since the control enters (3) through its endpoint, we can rewrite this standard condition as

$$d^*(s, g) = \tau + \inf_{\delta \in \mathcal{R}_\tau(s)} d^*(s + \delta, g). \tag{5}$$

This relation requires the greatest descent in remaining cost over horizon τ to equal τ. Our construction uses it to turn finite-horizon reachability into explicit supervision of the critic's local gradient.

Constraining the critic through a finite-horizon Hamiltonian. With (5) as our starting point, we seek to turn the relation into a tractable local training signal for the critic d_θ. Unfortunately, enforcing it directly requires solving a nonlinear optimization to minimize the critic over reachable endpoints for each sampled (s, g) pair (Rawlings et al., 2020). Instead, we simplify this inner optimization by *linearizing only the critic's dependence on its endpoint* as

$$d_\theta(s + \delta, g) = d_\theta(s, g) + \nabla_s d_\theta(s, g)^\top \delta + o(\|\delta\|).$$

Crucially, this first-order approximation acts on the critic, while $\mathcal{R}_\tau(s)$ still contains the endpoints generated by composed controls. Thus, a gradient evaluated at a single state can receive supervision along directions that become reachable through control composition. At states where $d_\theta(\cdot, g)$ is differentiable, this approximation gives our *finite-horizon Hamiltonian*

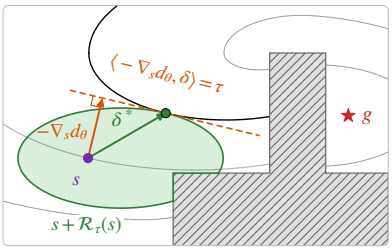

Figure 3: Toy illustration of (7), with $d_\theta \approx d^*$ in value and gradient and d^* affine along $[s, s + \delta^*]$. The orange line and black cost contour approximately touch the green reachable set at $s + \delta^*$. Contours are spaced by τ.

$$H_\tau(s, \nabla_s d_\theta(s, g)) := \frac{1}{\tau} \sup_{\delta \in \mathcal{R}_\tau(s)} \langle -\nabla_s d_\theta(s, g), \delta \rangle - 1. \tag{6}$$

As in classical minimum-time control, the Hamiltonian balances predicted cost descent against the running cost (Bardi et al., 1997). Similarly here, matching the predicted descent to the elapsed cost in (5) gives the first-order Bellman surrogate

$$H_\tau(s, \nabla_s d_\theta(s, g)) = 0 \quad \Longleftrightarrow \quad \sup_{\delta \in \mathcal{R}_\tau(s)} \langle -\nabla_s d_\theta(s, g), \delta \rangle = \tau. \tag{7}$$

This expresses the descent condition in (5) under the critic's first-order approximation. Going back to the example, the control sequence and its reverse at $s = 0$ produce sideways motions $(0, 0, \pm\tau^2/16)$ over duration τ. Since (7) requires the predicted descent along every reachable motion to be at most τ, these motions imply

$$H_\tau(0, \nabla_s d_\theta(0, g)) + 1 \geq \frac{\tau}{16} |\partial_z d_\theta(0, g)| \implies |\partial_z d_\theta(0, g)| \leq \frac{16}{\tau}.$$

The critic's sideways gradient component thus enters the finite-horizon constraint, together with the other two components through X and Y. Therefore, at $s = 0$, *all three components of* $\nabla_s d_\theta$ *enter the local constraint through motions reachable over a horizon* $\tau > 0$, provided $d_\theta(\cdot, g)$ is differentiable there (our critic is almost everywhere differentiable). Through $\mathcal{R}_\tau(s)$, we can supply the critic with explicit information about both local control dynamics and feasibility when learning the global goal-reaching geometry. Notice, however, that $\mathcal{R}_\tau(s)$ combines two distinct sources of structure: (i) what the controls can generate and (ii) which of those motions actually remain feasible within the domain. In the next section, we separate their roles in the critic constraint.

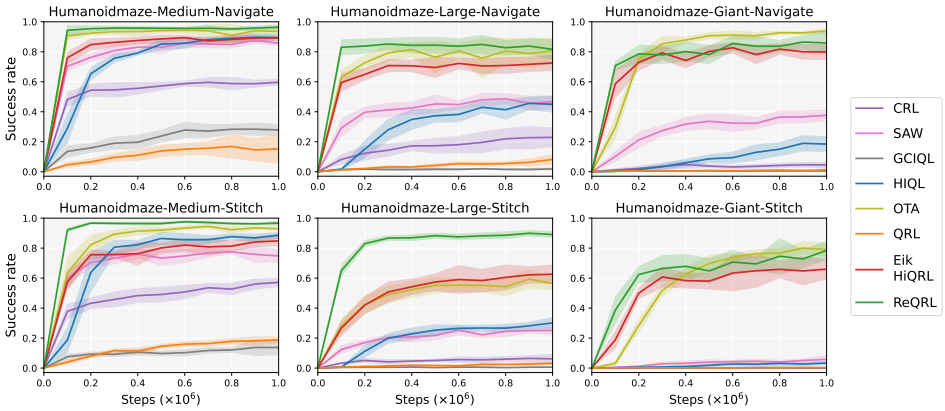

Figure 4: Evaluation curves on `humanoidmaze` for `navigate` and `stitch` datasets of all three sizes. The success rate with std shading is plotted every 10^5 training steps up to 10^6.

4 Decoupling: Dynamics in Interior, Feasibility at Boundary

Our formulation above combines two questions: **(i)** what state changes can the controls generate over duration τ and **(ii)** which trajectories remain feasible in $\mathcal{S}$? Estimating this jointly from data requires accounting for how composed controls interact with environmental constraints over the full horizon (Tebjou et al., 2023). State-constrained Hamilton-Jacobi theory (Soner, 1986; Aubin et al., 2011) suggests a natural way to approach this difficulty: the ambient control dynamics give the interior Hamiltonian, and the geometry of the feasible domain shows up via a boundary viability condition. Following this structure, we approximate the finite-horizon reachability constraints using separate estimates of dynamical reachability for interior constraints and domain geometry for the boundary constraint. We will assume the underlying dynamics are defined independently of $\mathcal{S}$.

Setting up dynamical reachability. Let $\mathcal{U}_\tau^{\mathrm{dyn}}(s)$ denote the controls over duration τ using actions in $\mathcal{A}$ without requiring the trajectory to remain feasible in $\mathcal{S}$. Let us write $\mathcal{T}_\tau^{\mathrm{dyn}}(s,u) \in \mathbb{R}^n$ for its endpoint. Every feasible control is dynamically admissible, so $\mathcal{U}_\tau(s) \subseteq \mathcal{U}_\tau^{\mathrm{dyn}}(s)$. Under the stated assumption above, we have $\mathcal{T}_\tau^{\mathrm{dyn}}(s,u) = \mathcal{T}_\tau(s,u)$ for $u \in \mathcal{U}_\tau(s)$. Define

$$\mathcal{R}_\tau^{\mathrm{dyn}}(s) := \left\{ \mathcal{T}_\tau^{\mathrm{dyn}}(s,u) - s : u \in \mathcal{U}_\tau^{\mathrm{dyn}}(s) \right\}. \tag{8}$$

This set describes what control dynamics permit over the horizon τ independently of the domain geometry, allowing the estimation of dynamical reachability and boundary feasibility separately. Assuming $\mathcal{R}_\tau^{\mathrm{dyn}}(s)$ is nonempty and compact, define the *dynamical finite-horizon Hamiltonian* as

$$H_\tau^{\mathrm{dyn}}(s,p) := \frac{1}{\tau} \sup_{\delta \in \mathcal{R}_\tau^{\mathrm{dyn}}(s)} \langle -p, \delta \rangle - 1, \qquad p \in \mathbb{R}^n.$$

In Soner (1986), the unconstrained dynamics determine the *interior* equation, while the state constraint enters through the *boundary*. Following the interior part of the state-constrained Hamilton-Jacobi structure, we impose the interior constraint $H_\tau^{\mathrm{dyn}}(s, \nabla_s d_\theta(s,g)) = 0$, or equivalently

$$\sup_{\delta \in \mathcal{R}_\tau^{\mathrm{dyn}}(s)} \langle -\nabla_s d_\theta(s,g), \delta \rangle = \tau, \qquad s \in \mathcal{S}^\circ \setminus \{g\}. \tag{9}$$

Setting up boundary feasibility. We incorporate domain geometry with a separate boundary condition. At a regular point $s \in \partial\mathcal{S}$, let $\nu(s)$ denote the outward unit normal. For solutions C^1 up to the boundary, the state-constraint condition of Soner (1986) requires that adding any nonnegative outward-normal component to the covector cannot decrease the Hamiltonian. Adapting this gives $H_\tau^{\mathrm{dyn}}(s,p) \le H_\tau^{\mathrm{dyn}}(s, p + \beta\nu(s))$ for all $\beta \ge 0$ with $p = \nabla_s d_\theta(s,g)$. Assuming $\mathcal{R}_\tau^{\mathrm{dyn}}(s)$ is compact and its support maximizer δ^* is unique, convexity of the Hamiltonian in p makes the above equivalent to

$$\nu(s)^\top \delta^* \le 0, \qquad s \in \partial\mathcal{S}. \tag{10}$$

Thus, the reachable motion maximizing the critic's predicted descent must be tangent to or point into the domain. For the isotropic case $\mathcal{R}_\tau^{\mathrm{dyn}}(s) = \tau\mathbb{B}_2$, with $\mathbb{B}_2$ the closed unit Euclidean ball, and

$p \neq 0$, we have $\delta^* = -\tau p/\|p\|_2$, so (10) reduces to

$$\nu(s)^\top \nabla_s d_\theta(s, g) \geq 0. \tag{11}$$

Remark. We account for feasibility through a local boundary condition, while (7) uses reachability that respects constraints over the full local horizon τ. Under the stated assumptions, the constrained and unconstrained maxima agree whenever an unconstrained maximizing control remains feasible throughout that horizon. The approximation is practically motivated when trajectories remain close to their starting states relative to local geometric scales. We formalize the conditions for agreement and the possible gap in Appendix B, and Section 6 shows its effectiveness in practice.

Critic learning objective. Putting these two together, while we retain QRL's quasimetric parameterization and distance-maximization objective (Wang et al., 2023), we obtain our critic objective:

$$\begin{aligned} \max_\theta \ \mathbb{E}_{s,g}[d_\theta(s, g)] \quad \text{s.t.} \quad & H_\tau^{\mathrm{dyn}}(s, \nabla_s d_\theta(s, g)) = 0, \quad s \in \mathcal{S}^\circ \setminus \{g\}, \\ & \nu(s)^\top \nabla_s d_\theta(s, g) \geq 0, \quad s \in \partial\mathcal{S} \setminus \{g\}. \end{aligned} \tag{12}$$

We use the isotropic boundary form as a geometric regularizer discouraging outward Euclidean descent without estimating a maximizing motion at the boundary. This simplifies boundary enforcement while retaining anisotropic reachability in the interior. In practice, we use soft constraint penalties and QRL's monotone distance transform (Appendix C). Section 5 describes our offline instantiation and estimation of dynamical reachability and boundary geometry.

5 From Transition Data to Critic Constraints

In deterministic GCRL, the dataset $\mathcal{D}$ contains transitions (s, a, s') with $s' = \mathcal{T}(s, a)$ at a fixed, unknown interaction interval $\Delta t > 0$. To work directly with this data, we instantiate the horizon as $\tau := k\Delta t$ with $k \in \mathbb{N}$. Controls over this duration are action sequences $u = (a_0, \ldots, a_{k-1}) \in \mathcal{A}^k$, with endpoints obtained by composing $\mathcal{T}$ for k environment steps. This uses the transition map directly without assuming a continuous-time generator, which need not exist or be unique (Palis, 1974; Yue et al., 2016). This k-step composition is related to temporal abstraction (Sutton et al., 1999); here, the composition defines the reachable set used in the finite-horizon critic constraint.

Reachability and cost normalization. At $\tau = k\Delta t$, we write $\mathcal{R}_k^{\mathrm{dyn}}(s) := \mathcal{R}_{k\Delta t}^{\mathrm{dyn}}(s)$ for the k-step reachable sets. We measure goal-reaching costs in units of the fixed horizon τ, dividing every policy's accumulated time cost by the same positive constant, preserving the set of optimal policies exactly. With d_θ denoting the normalized critic, the objective in (12) has the equivalent form

$$\begin{aligned} \max_\theta \ \mathbb{E}_{s,g}[d_\theta(s, g)] \quad \text{s.t.} \quad & \max_{\delta \in \mathcal{R}_k^{\mathrm{dyn}}(s)} \langle -\nabla_s d_\theta(s, g), \delta \rangle = 1, \quad s \in \mathcal{S}^\circ \setminus \{g\}, \\ & \nu(s)^\top \nabla_s d_\theta(s, g) \geq 0, \quad s \in \partial\mathcal{S} \setminus \{g\}. \end{aligned} \tag{13}$$

Positive scaling preserves the quasimetric structure and the boundary condition, and this formulation can be evaluated even when the numerical value of Δt is unknown. In our instantiation, we parameterize the critic using goal-relevant coordinates $x_s \in \mathbb{R}^{n_x}$ chosen offline, writing $d_\theta(s, g) = d_\theta(x_s, x_g)$. This restricts the approximation of the full-state value. To estimate reachable motions in these coordinates, we use a richer reduced state $r_s \in \mathbb{R}^{n_r}$ containing x_s and relevant dynamical variables, and then project the predicted motions onto the critic coordinates. We estimate dynamical reachability and boundary geometry from $\mathcal{D}$ using an affine dynamics model and empirical state occupancy, respectively. See Appendix C for full details.

Estimating dynamical reachability. We fit an affine model of the dynamics by ordinary least squares on offline transitions $(s, a, s') \in \mathcal{D}$ (Ljung, 1999; Verhaegen & Verdult, 2007): $r_{s'} = A_r r_s + B_r a + b_r + w$, with $a \in \mathcal{A} = [-1, 1]^m$, where w represents affine-model discrepancies, including nonlinear effects and omitted state dependence, and ranges over an empirical zonotopic error envelope estimated from residuals. Propagating this model for k steps and retaining the goal-coordinate displacements (Girard, 2005; Althoff et al., 2021) yields the estimate

$$\widehat{\mathcal{R}}_k^{\mathrm{dyn}}(s) = \{c_s + G_k z : \|z\|_\infty \leq 1\}. \tag{14}$$

We obtain c_s and G_k by initializing $c = r_s$ with no generators and iterating $c \leftarrow A_r c + b_r$ and $G \leftarrow [A_r G, B_r, G_w^\top]$ for k steps, where the error envelope is $\{G_w^\top \xi : \|\xi\|_\infty \leq 1\}$. We retain the

Env	Data	Size	Other GCRL						Quasimetric		
			CRL	SAW	GCIQL	HIQL	OTA	GAS	QRL	EikHiQRL	**ReQRL**
pointmaze	nav	medium	65 ± 10	$\mathbf{99 \pm 2}$	57 ± 1	92 ± 3	$\mathbf{94 \pm 3}$	75 ± 12	79 ± 9	82 ± 9	$\mathbf{94 \pm 7}$
		large	50 ± 12	60 ± 14	33 ± 2	49 ± 12	$\mathbf{97 \pm 3}$	72 ± 9	88 ± 9	83 ± 13	$\mathbf{98 \pm 3}$
		giant	26 ± 10	12 ± 14	0 ± 0	6 ± 7	83 ± 11	$\mathbf{86 \pm 8}$	73 ± 7	71 ± 18	$\mathbf{91 \pm 8}$
	sti	medium	2 ± 3	70 ± 14	30 ± 5	68 ± 12	81 ± 7	85 ± 11	80 ± 8	87 ± 12	$\mathbf{93 \pm 10}$
		large	0 ± 0	12 ± 10	27 ± 7	12 ± 8	37 ± 14	87 ± 13	82 ± 8	81 ± 11	$\mathbf{96 \pm 8}$
		giant	0 ± 0	2 ± 6	0 ± 0	0 ± 0	9 ± 10	71 ± 22	59 ± 10	60 ± 20	$\mathbf{81 \pm 15}$
antmaze	nav	medium	76 ± 5	$\mathbf{93 \pm 3}$	58 ± 9	$\mathbf{91 \pm 3}$	$\mathbf{91 \pm 1}$	$\mathbf{96 \pm 1}$	83 ± 9	$\mathbf{95 \pm 2}$	$\mathbf{94 \pm 5}$
		large	58 ± 7	73 ± 2	20 ± 4	72 ± 4	$\mathbf{91 \pm 2}$	$\mathbf{92 \pm 3}$	65 ± 18	83 ± 3	$\mathbf{95 \pm 1}$
		giant	5 ± 3	11 ± 4	0 ± 0	7 ± 4	35 ± 3	47 ± 25	0 ± 0	56 ± 9	$\mathbf{77 \pm 6}$
	sti	medium	33 ± 5	72 ± 3	38 ± 4	74 ± 5	$\mathbf{95 \pm 2}$	$\mathbf{97 \pm 1}$	68 ± 8	$\mathbf{94 \pm 2}$	$\mathbf{96 \pm 1}$
		large	7 ± 4	23 ± 5	6 ± 3	15 ± 4	59 ± 11	$\mathbf{92 \pm 2}$	13 ± 3	81 ± 6	$\mathbf{89 \pm 3}$
		giant	0 ± 0	0 ± 0	0 ± 0	0 ± 0	3 ± 1	$\mathbf{72 \pm 6}$	0 ± 0	41 ± 16	$\mathbf{67 \pm 6}$
	exp	medium	4 ± 4	31 ± 9	12 ± 4	30 ± 14	90 ± 12	$\mathbf{96 \pm 3}$	1 ± 2	45 ± 21	$\mathbf{97 \pm 4}$
		large	0 ± 0	4 ± 2	0 ± 1	1 ± 2	79 ± 12	$\mathbf{92 \pm 4}$	0 ± 0	9 ± 9	83 ± 6
humanoid	nav	medium	60 ± 2	86 ± 2	28 ± 4	89 ± 2	$\mathbf{94 \pm 3}$	$\mathbf{94 \pm 6}$	15 ± 10	89 ± 2	$\mathbf{96 \pm 2}$
		large	23 ± 7	47 ± 5	2 ± 1	45 ± 6	81 ± 9	$\mathbf{87 \pm 5}$	8 ± 4	73 ± 5	$\mathbf{82 \pm 7}$
		giant	5 ± 2	38 ± 4	0 ± 1	18 ± 5	$\mathbf{95 \pm 2}$	86 ± 3	1 ± 1	80 ± 5	86 ± 8
	sti	medium	57 ± 3	75 ± 4	14 ± 5	89 ± 2	$\mathbf{93 \pm 1}$	$\mathbf{98 \pm 2}$	19 ± 2	85 ± 4	$\mathbf{97 \pm 2}$
		large	6 ± 3	25 ± 4	1 ± 0	30 ± 4	56 ± 5	82 ± 5	3 ± 1	63 ± 7	$\mathbf{89 \pm 2}$
		giant	0 ± 0	6 ± 3	0 ± 0	3 ± 1	$\mathbf{79 \pm 5}$	$\mathbf{83 \pm 3}$	0 ± 0	66 ± 7	$\mathbf{79 \pm 6}$

Table 1: **Success rate %** (↑), percentage of episodes in which agent reaches goal g from the starting point s (mean $\pm$ std over 10 seeds of the final evaluation). The agent is tested for 50 episodes per goal. We **bold** the results within 95% of the best and underline the best.

goal-coordinate entries and rows and subtract x_s from the center to obtain displacements. z stacks the action and error coefficients across these steps. This propagation captures multistep directions expressible through the fitted affine model, but is an approximation to the general finite-horizon reachable set. The resulting set is a *zonotope*, whose support function has a closed form (Althoff & Frehse, 2016; Althoff et al., 2021). Thus, the estimated interior constraint in (13) becomes

$$-\langle \nabla_{x_s} d_\theta(x_s, x_g), c_s \rangle + \|G_k^\top \nabla_{x_s} d_\theta(x_s, x_g)\|_1 = 1. \tag{15}$$

We can evaluate this during training without enumerating action sequences or reachable-set vertices.

Estimating boundary geometry. We estimate goal-coordinate boundary geometry by rasterizing offline observations of x_s and applying morphological closing (Haralick et al., 1987). From this occupancy region, we construct a smooth implicit representation Φ of the estimated boundary, positive inside and negative outside, which gives $\widehat{\nu}(x_s) = -\nabla\Phi(x_s)/\|\nabla\Phi(x_s)\|_2$. This follows the level-set characterization of boundary normals (Delfour & Zolésio, 2011). We enforce the boundary penalty near the estimated boundary, assuming adequate coverage of the feasible region.

Finite time scale k. The horizon $k \geq 1$ sets the duration $\tau = k\Delta t$ over which controls are composed. We choose k by comparing the extent of the control-generated part of G_k with the nominal center displacement c_s in the goal coordinates. The control-generated part comes from the propagated generators $B_r, A_r B_r, \ldots, A_r^{k-1} B_r$ (Chen, 1984; Althoff et al., 2021). We choose a horizon at which these two contributions are comparable, so that composed actions can affect the goal coordinates. We fix k within each environment family across dataset types and domain sizes.

6 Experiments

Policy learning and control. We train an IQE critic using soft penalties for the normalized objective in (13) (Wang & Isola, 2022). The reachable set and boundary normals follow Section 5, with the interior penalty based on the zonotopic support in (15). Our critic receives *local training signals from motions composed over $\tau = k\Delta t$*. Thus, we supervise an actor at the same horizon, training an actor $\pi_\psi(w \mid x_s, x_g)$ on k-step hindsight *waypoints* $w = x_{s_{t+k}}$, truncated at trajectory ends. This aligns waypoint targets with the critic's reachability horizon and learned geometry. The actor predicts an offset from x_s and is trained by capped AWR (Peng et al., 2019) following HIQL (Park et al., 2023), using advantage $A^h(s, w, g) = d_\theta(x_s, x_g) - d_\theta(w, x_g)$. We supply these coordinate waypoints directly to the low-level policy, without any learned waypoint embedding. For $k > 1$, a separate

value function $V_\eta(s, w)$ takes the full observation and waypoint and is trained by action-free, goal-conditioned expectile TD learning (Kostrikov et al., 2021). The low-level policy $\pi_\ell(a \mid s, w)$ is then trained by AWR on recorded actions using HIQL's simplified advantage $V_\eta(s', w) - V_\eta(s, w)$. Full training and implementation details appear in Appendix C.

6.1 Main Benchmarks

Benchmarks and baselines. We evaluate on `OGBench`'s `pointmaze`, `antmaze`, and `humanoidmaze` environments across datasets and sizes (Park et al., 2025). Maze sizes are `medium`, `large`, and `giant` and the three robots are a point, an ant, and a human. Training dataset types are `navigate`, `stitch`, and `explore`. An agent must navigate from its initial position to a specified goal. We call our method **ReQRL** (Reachable QRL). We primarily compare with QRL (Wang et al., 2023) and EikHiQRL (Giammarino & Qureshi, 2026) to assess our quasimetric critic contribution. We compare with GCIQL (Kostrikov et al., 2021), CRL (Eysenbach et al., 2022), HIQL (Park et al., 2023), SAW (Zhou & Kao, 2026), OTA (Ahn et al., 2026), and GAS (Baek et al., 2025). Our reproduced baseline results largely match `OGBench` and respective papers. See Appendix C.

Env	Data	Size	Other GCRL		Quasimetric		
			OTA	GAS	QRL	EikHiQRL	**ReQRL**
antmaze	nav	medium	3 ± 0	2 ± 0	5 ± 1	3 ± 0	1 ± 0
		large	3 ± 0	3 ± 1	7 ± 2	5 ± 0	2 ± 0
		giant	9 ± 1	8 ± 3	14 ± 1	6 ± 1	4 ± 1
	sti	medium	3 ± 0	2 ± 0	6 ± 1	4 ± 0	1 ± 0
		large	8 ± 1	3 ± 1	10 ± 1	5 ± 1	1 ± 0
		giant	13 ± 1	5 ± 1	16 ± 1	7 ± 1	4 ± 1
	exp	medium	2 ± 1	4 ± 1	21 ± 2	13 ± 3	1 ± 1
		large	5 ± 2	5 ± 1	19 ± 1	11 ± 2	1 ± 1
humanoid	nav	medium	8 ± 1	7 ± 2	16 ± 4	9 ± 1	3 ± 0
		large	9 ± 1	9 ± 1	18 ± 3	13 ± 2	10 ± 3
		giant	9 ± 1	11 ± 1	25 ± 2	11 ± 1	7 ± 1
	sti	medium	7 ± 0	6 ± 1	15 ± 2	10 ± 1	4 ± 1
		large	10 ± 1	9 ± 1	18 ± 2	13 ± 1	6 ± 0
		giant	12 ± 1	13 ± 1	27 ± 2	13 ± 1	7 ± 1

Table 2: **Collision rate (%)** (↓): percentage of time steps with agent-boundary contact in `MuJoCo` (mean ± std over 10 seeds). Full results appear in Appendix D.

Training and evaluation. ReQRL and EikHiQRL share the IQE critic, AWR actor, and low-level TD critic architectures, with common hyperparameters matched except for the method-specific differences mentioned in Appendix C (along training budgets, hyperparameters, evaluation counts, and metric definitions). We report success and wall-collision rates on 10 seeds with their mean and std.

Comparison to quasimetric baselines. In Table 1, ReQRL obtains higher success than QRL and EikHiQRL in almost all settings. For instance, on `humanoid-large-stitch`, ReQRL reaches 89% while QRL and EikHiQRL reach 3% and 63% respectively. On `antmaze-giant`, ReQRL improves on EikHiQRL by 21 points on `navigate` and 26 points on `stitch`, and by 74 points on `antmaze-large-explore`. Critic constraints in ReQRL are computed from the identified reachability (i.e., control dynamics and boundaries) of the data without requiring complete successful trajectories, which is consistent with its robustness in `explore`.

Comparison to other GCRL methods. The competitive non-quasimetric methods are GAS and OTA, both recent state of the art on `OGBench`. Overall, ReQRL is comparable to or better than both. ReQRL is largely higher on `pointmaze` datasets and `antmaze-giant-navigate`, while GAS is higher on `antmaze-large-explore` and OTA on `humanoid-giant-navigate`. ReQRL thus closes the gap between quasimetric methods and the state of the art in offline GCRL. Figure 7 further shows ReQRL on the Pareto frontier of training time and success.

Convergence and collision rate. Figure 4 shows training curves on `humanoidmaze`, where ReQRL learns quickly and reaches the best or near-best final success, consistent with the explicit reachability information supplied to the critic. We also study boundary interactions during policy execution through Table 2. ReQRL achieves lower mean collision rates than QRL and EikHiQRL, and lower or comparable collision rates compared with OTA and GAS across the reported `antmaze` and `humanoidmaze` settings. Full results, including mixed `pointmaze`, are in Appendix D.

6.2 Nonholonomic Control Dynamics

We additionally test our finite-horizon value supervision under nominally *nonholonomic* dynamics, where orientation constrains translation (i.e., turning rather than stepping sideways). This provides a targeted setting for evaluating our approach when reachability depends on composing turning and translation. To this end, we construct the `carmaze` environment (Figure 5) by integrating the `Safety-Gymnasium Car` robot (Ji et al., 2023), natively simulated in `MuJoCo` (Todorov et al., 2012), into the same `OGBench` layouts. Following its collection pipeline, an SAC locomotion expert

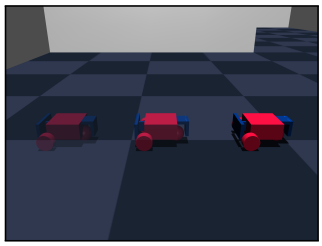

Figure 5: `Safety-Gymnasium Car` moving in `OGBench`'s maze.

Env	Data	Size	Other GCRL		Quasimetric	
			OTA	GAS	EikHiQRL	**ReQRL**
carmaze	nav	medium	63 ± 7	81 ± 5	72 ± 8	$\mathbf{90 \pm 3}$
		large	32 ± 14	67 ± 7	45 ± 15	$\mathbf{81 \pm 12}$
		giant	3 ± 3	40 ± 8	13 ± 9	$\mathbf{60 \pm 14}$
	sti	medium	60 ± 6	74 ± 6	69 ± 9	$\mathbf{82 \pm 6}$
		large	14 ± 6	64 ± 10	26 ± 9	$\mathbf{78 \pm 12}$
		giant	0 ± 0	31 ± 8	5 ± 7	$\mathbf{45 \pm 17}$

Table 3: **Success rates (%) (↑)** on `carmaze` (mean ± std over 10 seeds). Full results appear in Appendix D.

with waypoint guidance generates `navigate` and `stitch` datasets. All methods share the datasets and evaluation protocol; car-specific adaptations and tuning details are provided in Appendix D.

Success rate in `carmaze`. In Table 3, ReQRL outperforms baselines in all settings. On giant `navigate` and `stitch`, it reaches 60% and 45%, versus 40% and 31% for GAS. These gains are consistent with our motivation, where finite-horizon supervision explicitly incorporates composed motions into local critic constraints when progress requires composing turning and translation. The `stitch` gains support its usefulness for reaching distant goals from short trajectory segments.

7 Related Work

Goal-conditioned RL. GCRL provides a framework for learning reusable policies across goals (Schaul et al., 2015; Pong et al., 2018) and turning achieved outcomes into self-supervision through hindsight relabeling (Andrychowicz et al., 2017; Ghosh et al., 2019). Offline GCRL extends this to reward-free datasets without further interaction (Chebotar et al., 2021; Park et al., 2025). Approaches include weighted imitation (Yang et al., 2022), policy regression via occupancy matching (Ma et al., 2022), and contrastive value learning (Eysenbach et al., 2022). To tackle longer horizons, others use subgoal reaching (Park et al., 2023), advantage estimates via temporal abstraction (Ahn et al., 2026), flat-policy bootstrapping (Zhou & Kao, 2026), and graph-based stitching (Baek et al., 2025). A complementary line exploits the metric and geometric structure of goal-reaching values.

(Quasi)metrics in GCRL. Viewing costs as distances motivates reward shaping via dynamical distances (Hartikainen et al., 2019) and subgoal search (Eysenbach et al., 2019), planning embeddings for navigation (Yang et al., 2020), and metric representations for skill discovery (Park et al., 2024). A key insight is that *quasimetrics* encode both directional costs and path composition. QRL turns this into a learning principle linking local transition constraints to global distances (Wang et al., 2023). CMD and TMD combine quasimetric structure with contrastive temporal-distance learning (Myers et al., 2024; 2026). EikHiQRL regularizes critic gradients with Eikonal constraints under simplified dynamics (Giammarino & Qureshi, 2026). Grounding local supervision in composed reachability and state feasibility remains challenging, motivating geometric and optimal control.

Geometric and optimal control. Geometric control shows composing controls can generate motion beyond the instantaneous span (Jean, 2014). In optimal control, Hamilton-Jacobi theory connects dynamics to value gradients (Bardi et al., 1997), while state-constraint boundary conditions encode feasibility (Soner, 1986). In RL, these ideas motivate HJB-based derivative regularization (Lien et al., 2024) and Eikonal-based TD learning (Giammarino et al., 2026). ReQRL unifies these in local supervision: a finite-horizon Hamiltonian preserves control composition while linearizing only the critic and boundary constraints encode feasibility.

8 Conclusion

We formulated finite-horizon Hamiltonian constraints for learning global goal-reaching geometry through local quasimetric critic supervision. Our approach retains the effects of controls composed over time through the reachable set. We further separate dynamical reachability from boundary geometry, providing complementary signals about control dynamics and feasibility. With this, we developed ReQRL for GCRL, with data-driven zonotopic reachability estimates and boundary regularization. Experiments on `OGBench` show improvements over quasimetric baselines and competitive performance against other GCRL methods. We discuss limitations in Appendix A.

AI USE STATEMENT

In this work, we used generative AI tools for refining writing at the word, sentence, and paragraph level, to aid in the generation of figures, to aid in literature search, editing the presentation of the proofs we derived, and to aid in coding with LLM agents. We have not used generative AI tools for proving mathematical claims or writing beyond the paragraph scale. We have reviewed all AI-assisted work and have verified the correctness of writing and code. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

All experiments run in simulation in `MuJoCo`, using publicly released `OGBench` datasets and `carmaze` data collected with the `Safety-Gymnasium Car` model. No human subjects or personal information are involved. The contribution is a training signal for an offline critic, and we see no harms that are specific to it. The usual caution for reinforcement learning applies if controllers trained this way are ever moved onto physical robots, where they would need the same safety validation as any learned controller.

REPRODUCIBILITY STATEMENT

We provide the full method in Sections 4 and 5, along with training details, budgets, and hyperparameters in Section 6 and Appendix C and D. We provide our method's code in the supplementary material.

REFERENCES

Andrei Agrachev, Davide Barilari, and Ugo Boscain. *A comprehensive introduction to sub-Riemannian geometry*. Cambridge University Press, 2020.

Hongjoon Ahn, Heewoong Choi, Jisu Han, and Taesup Moon. Option-aware temporally abstracted value for offline goal-conditioned reinforcement learning. *Advances in Neural Information Processing Systems*, 38:99833–99861, 2026.

Matthias Althoff and Goran Frehse. Combining zonotopes and support functions for efficient reachability analysis of linear systems. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 7439–7446. IEEE, 2016.

Matthias Althoff, Goran Frehse, and Antoine Girard. Set propagation techniques for reachability analysis. *Annual Review of Control, Robotics, and Autonomous Systems*, 4(1):369–395, 2021.

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *Advances in neural information processing systems*, 30, 2017.

Jean-Pierre Aubin, Alexandre M. Bayen, and Patrick Saint-Pierre. *Viability Theory: New Directions*. Springer, 2 edition, 2011. doi: 10.1007/978-3-642-16684-6.

Seungho Baek, Taegeon Park, Jongchan Park, Seungjun Oh, and Yusung Kim. Graph-assisted stitching for offline hierarchical reinforcement learning. *arXiv preprint arXiv:2506.07744*, 2025.

Martino Bardi, Italo Capuzzo Dolcetta, et al. *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations*, volume 12. Springer, 1997.

Franco Blanchini, Stefano Miani, et al. *Set-theoretic methods in control*, volume 78. Springer, 2008.

Yevgen Chebotar, Karol Hausman, Yao Lu, Ted Xiao, Dmitry Kalashnikov, Jake Varley, Alex Irpan, Benjamin Eysenbach, Ryan Julian, Chelsea Finn, et al. Actionable models: Unsupervised offline reinforcement learning of robotic skills. *arXiv preprint arXiv:2104.07749*, 2021.

Chi-Tsong Chen. *Linear system theory and design*, volume 301. Holt, Rinehart and Winston New York, 1984.

Michel C Delfour and J-P Zolésio. *Shapes and geometries: metrics, analysis, differential calculus, and optimization*. SIAM, 2011.

Ben Eysenbach, Russ R Salakhutdinov, and Sergey Levine. Search on the replay buffer: Bridging planning and reinforcement learning. *Advances in neural information processing systems*, 32, 2019.

Benjamin Eysenbach, Tianjun Zhang, Sergey Levine, and Russ R Salakhutdinov. Contrastive learning as goal-conditioned reinforcement learning. *Advances in Neural Information Processing Systems*, 35:35603–35620, 2022.

Dibya Ghosh, Abhishek Gupta, Ashwin Reddy, Justin Fu, Coline Devin, Benjamin Eysenbach, and Sergey Levine. Learning to reach goals via iterated supervised learning. *arXiv preprint arXiv:1912.06088*, 2019.

Vittorio Giammarino and Ahmed Hussain Qureshi. Goal reaching with eikonal-constrained hierarchical quasimetric reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 59145–59182, 2026.

Vittorio Giammarino, Ruiqi Ni, and Ahmed Qureshi. Physics-informed value learner for offline goal-conditioned reinforcement learning. *Advances in Neural Information Processing Systems*, 38:137769–137796, 2026.

Antoine Girard. Reachability of uncertain linear systems using zonotopes. In *HSCC*, volume 3414, pp. 291–305, 2005.

Robert M Haralick, Stanley R Sternberg, and Xinhua Zhuang. Image analysis using mathematical morphology. *IEEE transactions on pattern analysis and machine intelligence*, (4):532–550, 1987.

Kristian Hartikainen, Xinyang Geng, Tuomas Haarnoja, and Sergey Levine. Dynamical distance learning for semi-supervised and unsupervised skill discovery. *arXiv preprint arXiv:1907.08225*, 2019.

Frédéric Jean. *Control of nonholonomic systems: from sub-Riemannian geometry to motion planning*. Springer, 2014.

Jiaming Ji, Borong Zhang, Jiayi Zhou, Xuehai Pan, Weidong Huang, Ruiyang Sun, Yiran Geng, Yifan Zhong, Josef Dai, and Yaodong Yang. Safety gymnasium: A unified safe reinforcement learning benchmark. *Advances in Neural Information Processing Systems*, 36:18964–18993, 2023.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169*, 2021.

Daniel Liberzon. Calculus of variations and optimal control theory: a concise introduction. 2011.

Yun-Hsuan Lien, Ping-Chun Hsieh, Tzu-Mao Li, and Yu-Shuen Wang. Enhancing value function estimation through first-order state-action dynamics in offline reinforcement learning. In *Forty-first International Conference on Machine Learning*, 2024.

Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. In *International Conference on Learning Representations*, 2016. URL https://arxiv.org/abs/1509.02971.

L. Ljung. *System Identification: Theory for the User*. Prentice Hall information and system sciences series. Prentice Hall PTR, 1999. ISBN 9780136566953. URL https://books.google.com/books?id=nHFoQgAACAAJ.

Yecheng Jason Ma, Jason Yan, Dinesh Jayaraman, and Osbert Bastani. How far i'll go: Offline goal-conditioned reinforcement learning via f-advantage regression. *arXiv preprint arXiv:2206.03023*, 2022.

Vivek Myers, Chongyi Zheng, Anca Dragan, Sergey Levine, and Benjamin Eysenbach. Learning temporal distances: Contrastive successor features can provide a metric structure for decision-making. *arXiv preprint arXiv:2406.17098*, 2024.

Vivek Myers, Bill Zheng, Benjamin Eysenbach, and Sergey Levine. Offline goal-conditioned reinforcement learning with quasimetric representations. *Advances in Neural Information Processing Systems*, 38:19654–19679, 2026.

Jacob Palis. Vector fields generate few diffeomorphisms. *Bulletin of the American Mathematical Society*, 80(3):503–505, 1974.

Seohong Park, Dibya Ghosh, Benjamin Eysenbach, and Sergey Levine. Hiql: Offline goal-conditioned rl with latent states as actions. *Advances in Neural Information Processing Systems*, 36:34866–34891, 2023.

Seohong Park, Tobias Kreiman, and Sergey Levine. Foundation policies with hilbert representations. *arXiv preprint arXiv:2402.15567*, 2024.

Seohong Park, Kevin Frans, Benjamin Eysenbach, and Sergey Levine. Ogbench: Benchmarking offline goal-conditioned rl. In *International Conference on Learning Representations*, volume 2025, pp. 94937–94982, 2025.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019.

Vitchyr Pong, Shixiang Gu, Murtaza Dalal, and Sergey Levine. Temporal difference models: Model-free deep rl for model-based control. *arXiv preprint arXiv:1802.09081*, 2018.

James B Rawlings, David Q Mayne, and Moritz M Diehl. Model predictive control: theory, computation, and design. *(No Title)*, 2020.

Tom Schaul, Daniel Horgan, Karol Gregor, and David Silver. Universal value function approximators. In *International conference on machine learning*, pp. 1312–1320. PMLR, 2015.

Halil Mete Soner. Optimal control with state-space constraint i. *SIAM Journal on Control and Optimization*, 24(3):552–561, 1986.

Richard S Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2):181–211, 1999.

Abdelmouaiz Tebjou, Goran Frehse, et al. Data-driven reachability using christoffel functions and conformal prediction. In *Conformal and Probabilistic Prediction with Applications*, pp. 194–213. PMLR, 2023.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pp. 5026–5033. IEEE, 2012.

Michel Verhaegen and Vincent Verdult. *Filtering and system identification: a least squares approach*. Cambridge university press, 2007.

Tongzhou Wang and Phillip Isola. Improved representation of asymmetrical distances with interval quasimetric embeddings. In *NeurIPS 2022 Workshop on Symmetry and Geometry in Neural Representations*, 2022. URL https://openreview.net/forum?id=KRiST_rzkGl.

Tongzhou Wang, Antonio Torralba, Phillip Isola, and Amy Zhang. Optimal goal-reaching reinforcement learning via quasimetric learning. In *International Conference on Machine Learning*, pp. 36411–36430. PMLR, 2023.

Ge Yang, Amy Zhang, Ari Morcos, Joelle Pineau, Pieter Abbeel, and Roberto Calandra. Plan2vec: Unsupervised representation learning by latent plans. In *Learning for Dynamics and Control*, pp. 935–946. PMLR, 2020.

Rui Yang, Yiming Lu, Wenzhe Li, Hao Sun, Meng Fang, Yali Du, Xiu Li, Lei Han, and Chongjie Zhang. Rethinking goal-conditioned supervised learning and its connection to offline rl. *arXiv preprint arXiv:2202.04478*, 2022.

Zuogon Yue, Johan Thunberg, and Jorge Gonçalves. Inverse problems for matrix exponential in system identification: system aliasing. *arXiv preprint arXiv:1605.06973*, 2016.

John Zhou and Jonathan Kao. Flattening hierarchies with policy bootstrapping. *Advances in Neural Information Processing Systems*, 38:82975–83007, 2026.

APPENDIX

A LIMITATIONS AND FUTURE WORK

We see several natural directions for extending our method. Our instantiation in Section 5 uses an affine dynamics model and occupancy-based boundary estimates to realize tractable evaluation of the constraints in (13). While these approximations are effective in our `OGBench` experiments, the affine model may not capture all state-dependent and nonlinear effects of the underlying control dynamics. The estimates also depend on coverage of relevant states in the training data $\mathcal{D}$, a concern shared by offline baselines such as QRL and EikHiQRL (Wang et al., 2023; Giammarino & Qureshi, 2026). Richer state representations and nonlinear reachability estimators could broaden applicability; an important challenge is retaining efficient constraint evaluation. Our formulation in Section 3 also assumes deterministic dynamics, and the experiments in Section 6 evaluate offline tasks. Extending the formulation to stochastic control systems and evaluating the method with online data collection are promising directions for future work.

B THEORETICAL ANALYSIS

We first show that expressing Bellman optimality through controls composed over duration τ preserves the target goal-reaching value d^* and its quasimetric structure. This provides the foundation for our finite-horizon critic constraints.

Lemma B.1 (Goal-reaching geometry under compositional control)**.** *Consider the minimum-cost d^*, $\mathcal{U}_\tau(s)$, and $\mathcal{T}_\tau$ from equation 3, and $\mathcal{R}_\tau(s)$ from equation 4. Assume d^* is finite and feasible controls can be joined and split at supported times. Then $d^* \in \mathcal{Q}(\mathcal{S})$ and satisfies equation 5 whenever $d^*(s, g) > \tau$. Moreover, d^* is the pointwise greatest quasimetric satisfying $d(s, \mathcal{T}_\tau(s, u)) \leq \tau$ for every supported $\tau > 0$, $s \in \mathcal{S}$, and $u \in \mathcal{U}_\tau(s)$.*

Proof. Nonnegativity and $d^*(s, s) = 0$ follow from the definition. Joining arbitrarily near-optimal controls through an intermediate state gives

$$d^*(s_1, s_3) \leq d^*(s_1, s_2) + d^*(s_2, s_3).$$

Since d^* is finite, $d^* \in \mathcal{Q}(\mathcal{S})$. Every $u \in \mathcal{U}_\tau(s)$ reaches $\mathcal{T}_\tau(s, u)$ in duration τ, so $d^*(s, \mathcal{T}_\tau(s, u)) \leq \tau$. Conversely, any quasimetric d satisfying these bounds obeys $d(s, g) \leq \tau$ whenever $\mathcal{T}_\tau(s, u) = g$. Taking the infimum over feasible goal-reaching controls gives $d(s, g) \leq d^*(s, g)$. For Bellman optimality, joining a control in $\mathcal{U}_\tau(s)$ with a near-optimal continuation gives

$$d^*(s, g) \leq \tau + \inf_{u \in \mathcal{U}_\tau(s)} d^*(\mathcal{T}_\tau(s, u), g).$$

Conversely, when $d^*(s, g) > \tau$, every near-optimal goal-reaching trajectory extends beyond τ. Splitting it there, its remaining duration is at least the optimal cost from $\mathcal{T}_\tau(s, u)$. Taking arbitrarily near-optimal trajectories gives the reverse inequality. Substituting equation 4 yields equation 5. □

Lemma B.1 establishes that finite-horizon reachability describes the same optimal goal-reaching value. We next connect this formulation to the transition data in Section 5 and show that the normalization used in equation 13 changes only the cost units.

Lemma B.2 (Data instantiation and cost normalization)**.** *Under the instantiation $\tau = k\Delta t$ in Section 5, with goal reaching evaluated at each environment step, dividing time costs by τ preserves optimal policies and expresses costs in environment steps divided by k. Scaling $d_\theta \mapsto d_\theta/\tau$ converts equation 12 to equation 13, preserving quasimetric structure and the boundary condition.*

Proof. Under the stated assumption that each action takes duration Δt, an action sequence of length k takes duration τ and reaches the endpoint obtained by composing $\mathcal{T}$ k times. The reachable-set identities follow. For any policy, a trajectory reaching the goal after N environment steps incurs time cost $N\Delta t$. Its normalized cost is

$$\frac{N\Delta t}{\tau} = \frac{N}{k}.$$

This positive scaling preserves policy ordering, including after taking expectations. It therefore preserves the optimal policies. Since τ is fixed,

$$H_\tau^{\mathrm{dyn}}(s, \nabla_s d_\theta(s,g)) = \max_{\delta \in \mathcal{R}_k^{\mathrm{dyn}}(s)} \left\langle -\nabla_s \left(\frac{d_\theta(s,g)}{\tau} \right), \delta \right\rangle - 1.$$

Positive scaling also preserves the quasimetric axioms, the boundary condition, and the ordering of the distance-maximization objective. Reusing d_θ for the normalized critic, as in Section 5, gives equation 13. The numerical value of Δt cancels. □

As discussed in Section 4, we now quantify the Hamiltonian gap introduced by replacing feasible reachability with dynamical reachability on the interior with a separate boundary condition.

Lemma B.3 (Dynamics-feasibility gap)**.** *Fix $s \in \mathcal{S}$ and $\tau > 0$. Assume the sets in equation 4 and equation 8 are nonempty and compact, with $\mathcal{R}_\tau(s) \subseteq \mathcal{R}_\tau^{\mathrm{dyn}}(s)$ as in our formulation. Define*

$$e_\tau(s) := \max_{\delta \in \mathcal{R}_\tau^{\mathrm{dyn}}(s)} \min_{\bar{\delta} \in \mathcal{R}_\tau(s)} \|\delta - \bar{\delta}\|_2.$$

Then, for every $p \in \mathbb{R}^n$,

$$0 \le H_\tau^{\mathrm{dyn}}(s,p) - H_\tau(s,p) \le \frac{\|p\|_2 e_\tau(s)}{\tau}.$$

The two Hamiltonians agree if and only if some maximizer over $\mathcal{R}_\tau^{\mathrm{dyn}}(s)$ belongs to $\mathcal{R}_\tau(s)$.

Proof. Set inclusion gives the lower bound. By compactness, choose a maximizing displacement $\delta^* \in \mathcal{R}_\tau^{\mathrm{dyn}}(s)$ and a nearest $\bar{\delta} \in \mathcal{R}_\tau(s)$. Then

$$\begin{aligned} \tau\big(H_\tau^{\mathrm{dyn}}(s,p) - H_\tau(s,p)\big) &= \langle -p, \delta^* \rangle - \max_{\delta \in \mathcal{R}_\tau(s)} \langle -p, \delta \rangle \\ &\le \langle -p, \delta^* - \bar{\delta} \rangle \\ &\le \|p\|_2 e_\tau(s). \end{aligned}$$

If an unconstrained maximizer belongs to the feasible set, the maxima agree. Conversely, if the maxima agree, any feasible maximizer, which exists by compactness, also maximizes over the unconstrained set. □

Here, $e_\tau(s)$ measures the largest distance from a dynamically reachable displacement to the feasible reachable set. The Hamiltonian gap vanishes whenever an unconstrained maximizer is feasible; otherwise, the bound quantifies its dependence on reachability mismatch, critic gradient, and horizon.

Theorem B.4 (Value approximation under true reachability)**.** *Under the assumptions of Lemma B.1, fix $g \in \mathcal{S}$ and a supported horizon $\tau > 0$. Assume each $\mathcal{R}_\tau(s)$ is nonempty and compact. Let $d_\theta(\cdot, g)$ be continuous, finite, and nonnegative, with $d_\theta(g,g) = 0$. Let $\mathcal{B}_g \subseteq \mathcal{S}$ contain $\{s : d^*(s,g) \le \tau\}$, and assume $|d_\theta(s,g) - d^*(s,g)| \le \beta$ on $\mathcal{B}_g$. Outside $\mathcal{B}_g$, assume the constraint in equation 7 holds up to residual*

$$|H_\tau(s, \nabla_s d_\theta(s,g))| \le \epsilon.$$

Assume also that $\|\delta\|_2 \le \rho$ for every $s \notin \mathcal{B}_g$ and $\delta \in \mathcal{R}_\tau(s)$, and that $d_\theta(\cdot, g)$ has an L-Lipschitz gradient on an open set containing every segment $[s, s+\delta]$ for these pairs. If $\beta, \epsilon, L, \rho \ge 0$ and $\epsilon + L\rho^2/(2\tau) < 1$, then, for every $s \in \mathcal{S}$,

$$|d_\theta(s,g) - d^*(s,g)| \le \left(\epsilon + \frac{L\rho^2}{2\tau} \right) d^*(s,g) + \beta.$$

Proof. Fix $g \in \mathcal{S}$, and write $f(s) = d_\theta(s,g)$ and $\eta = \epsilon + L\rho^2/(2\tau) < 1$. For $s \notin \mathcal{B}_g$ and $\delta \in \mathcal{R}_\tau(s)$, the Lipschitz-gradient assumption gives

$$|f(s+\delta) - f(s) - \nabla f(s)^\top \delta| \le \frac{L}{2}\|\delta\|_2^2 \le \frac{L\rho^2}{2}.$$

Taking maxima and using the Hamiltonian residual bound yields

$$\tau(1-\eta) \le f(s) - \min_{\delta \in \mathcal{R}_\tau(s)} f(s+\delta) \le \tau(1+\eta).$$

Continuity and compactness ensure that the minimum is attained.

For the upper bound, fix $s \notin \mathcal{B}_g$ and take any feasible goal-reaching trajectory of duration T. Sample it at times $0, \tau, 2\tau, \dots$ up to T, and let b be its first sampled state in $\mathcal{B}_g$, reached after N intervals. Such a state exists because the last sampled state has remaining goal-reaching time at most τ. Since $d^*(b, g) \le T - N\tau$, the upper inequality along these intervals gives

$$f(s) \le (1+\eta)N\tau + f(b) \le (1+\eta)N\tau + d^*(b,g) + \beta \le (1+\eta)T + \beta.$$

Taking the infimum over feasible durations T yields $f(s) \le (1+\eta)d^*(s,g) + \beta$.

For the lower bound, start at $s_0 = s$ and choose $s_{i+1} \in \arg\min_{y \in s_i + \mathcal{R}_\tau(s_i)} f(y)$ until entering $\mathcal{B}_g$. Before entry, $f(s_{i+1}) \le f(s_i) - \tau(1-\eta)$. Since $f \ge 0$ and $\tau(1-\eta) > 0$, entry must occur after finitely many intervals. Let b be the entry state after N intervals. Concatenation gives $d^*(s,g) \le N\tau + d^*(b,g)$, and hence

$$f(s) \ge (1-\eta)N\tau + f(b) \ge (1-\eta)N\tau + d^*(b,g) - \beta \ge (1-\eta)d^*(s,g) - \beta.$$

Combining the bounds proves the claim outside $\mathcal{B}_g$. On $\mathcal{B}_g$, it follows directly from the assumed error bound. □

We note that the above theorem provides global value error bounds in terms of the constraint residual, the first-order approximation error, and the assumed accuracy on $\mathcal{B}_g$. While this bound gives some insight for the value recovery of our method in the idealized setting, this analysis should thus be interpreted only as conditional guarantee for the true reachable set. The practical feasibility and effectiveness of the method are more directly established via our experiments in Section 6.

C Main Experiment Details

C.1 Benchmarks

Benchmarks and datasets. We use the state-based environments from `OGBench` (Park et al., 2025), simulated in `MuJoCo` (Todorov et al., 2012). Our main experiments cover `pointmaze`, `antmaze`, and `humanoidmaze` across `medium`, `large`, and `giant` layouts.

In `pointmaze`, a point mass moves through the maze, with its 2-D position as the state and 2-D actions directly controlling planar movement. In `antmaze`, a quadrupedal ant walks toward the goal using 8 joint-torque controls and a 29-D state. In `humanoidmaze`, a humanoid must coordinate balance and walking using 21 joint-torque controls and a 69-D state. The latter two states include position, body configuration, and motion information. All action spaces are boxes, $\mathcal{A} = [-1, 1]^m$, where m is the action dimension. We use the officially released `navigate` and `stitch` training datasets for all three families, and `explore` for `antmaze-medium` and `antmaze-large`. The `navigate` datasets contain trajectories pursuing randomly sampled goals, `stitch` contains short goal-reaching trajectories requiring composition to solve longer tasks, and `explore` contains noisy exploratory trajectories generated from random directional commands. Table 4 summarizes observation and action dimensions and training dataset sizes. All methods use the same training datasets and number of training updates within each setting. Appendix D additionally reports results on `antsoccer-arena-navigate` and `antsoccer-medium-navigate`. These tasks require an ant to approach and dribble a ball to a goal location, in an open arena or a maze, respectively. Success depends on the ball's position.

Environment	n	m	Training transitions per dataset
`pointmaze`	2	2	1M (`navigate`, `stitch`)
`antmaze`	29	8	1M (`navigate`, `stitch`); 5M (`explore`)
`humanoidmaze`	69	21	2M (`medium`, `large`); 4M (`giant`)

Table 4: Observation/action dimensions and training dataset sizes. M denotes one million transitions; counts exclude terminal-only observations. n and m denote the full state and action spaces, respectively. The `humanoidmaze` counts apply to both dataset types.

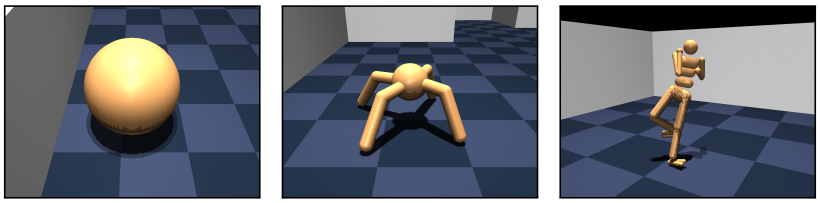

Figure 6: The `pointmaze`, `antmaze`, and `humanoidmaze` agents of `OGBench`.

Evaluations. We evaluate all the methods on the five prescribed `OGBench` tasks, with 50 episodes per task and the benchmark's success criterion and episode limits, following the standard evaluation protocol. Success rates (%) are averaged across tasks for each training seed, and we report the mean and standard deviation over 10 seeds. Collision rates (%) measure the percentage of evaluation steps with at least one `MuJoCo` contact between the agent and the domain boundary. Exact training budgets and hyperparameters are provided below.

C.2 Reachability and Boundary Estimation

State representations. For all three environment families, the critic d_θ uses the planar physical position $x_s = (x, y) \in \mathbb{R}^2$ of the full $\mathcal{S}$, with goals represented by the corresponding coordinates x_g, as we mention in Section 5. To estimate $\mathcal{R}_\tau^{\mathrm{dyn}}(s)$, we use reduced states r_s containing these coordinates and additional dynamical variables. For `pointmaze`, $r_s = (x, y) \in \mathbb{R}^2$. For `antmaze`, $r_s = (x, y, v_x, v_y, \cos\psi, \sin\psi) \in \mathbb{R}^6$, where (v_x, v_y) are planar translational velocities and ψ is the heading angle obtained from the body's orientation quaternion. For `humanoidmaze`, $r_s = (x, y, v_x, v_y, o_1, o_2, o_3) \in \mathbb{R}^7$, where (v_x, v_y) are root planar velocities and (o_1, o_2, o_3) encode the world vertical direction in the torso frame. We propagate the dynamics in these reduced states and retain the (x, y) displacements for the critic constraint. These coordinates and dynamical variables are selected using only offline observations, without additional environment interaction. We also use x_s directly as the waypoint representation, avoiding the separate learned waypoint embedding used in HIQL and EikHiQRL (Park et al., 2023; Giammarino & Qureshi, 2026).

Estimating the local constraint geometry. To estimate $\mathcal{R}_\tau^{\mathrm{dyn}}(s)$,, for each environment family and dataset type of `OGBench`, we fit one reduced dynamics model a priori using *only* the training dataset $\mathcal{D}$ and fix it throughout the training of the critic d_θ. We form one-step pairs $(r_s, a, r_{s'})$ within each recorded trajectory $(s, a, s') \in \mathcal{D}$, excluding pairs that cross episode cutoffs. Then, we estimate the affine model parameters by minimizing ordinary least squares (OLS) (Ljung, 1999; Verhaegen & Verdult, 2007):

$$(A_r, B_r, b_r) \in \arg\min_{A,B,b} \sum_{(s,a,s')} \|r_{s'} - Ar_s - Ba - b\|_2^2.$$

For residuals $e_i = r_{s_i'} - A_r r_{s_i} - B_r a_i - b_r$, we then compute

$$C = \frac{1}{N}\sum_{i=1}^{N} e_i e_i^\top = Q\Lambda Q^\top, \qquad G_w = 2\Lambda^{1/2} Q^\top.$$

The empirical error envelope is given by $\{G_w^\top \xi : \|\xi\|_\infty \le 1\}$, with zonotope generators spanning two empirical standard deviations along each principal residual direction. For `pointmaze`, we fit transitions unaffected by boundary contact. At horizon $\tau = k\Delta t$, the propagated center in the reduced state space is obtained as

$$A_r^k r_s + \sum_{j=0}^{k-1} A_r^j b_r,$$

and the generators comprise the action blocks $A_r^{k-1}B_r, \ldots, B_r$ and error blocks $A_r^{k-1}G_w^\top, \ldots, G_w^\top$, with coefficients ranging independently over $[-1, 1]$ (Girard, 2005; Althoff et al., 2021). We retain the planar-position entries of the center and corresponding generator rows, then subtract x_s from the center to obtain c_s and G_k in equation 14. The generator matrix is shared across states for each fitted model and horizon, while c_s depends on the current reduced state. These quantities are precomputed where possible, and the critic constraint is evaluated through the closed-form support in equation 15.

To estimate boundary geometry and outward unit normals, we use grid resolutions of at most 0.5 and one or two morphological closing iterations with a four-neighbor structuring element for each layout. We smooth the signed distance field with a Gaussian of standard deviation one grid cell, compute gradients by central differences, and exclude points with $\|\nabla\Phi\|_2 \le 10^{-2}$.

To determine the horizon $\tau := k\Delta t$ before critic training, we heuristically compare the control-generated part of G_k with the uncontrolled displacement c_s, both obtained from the dynamics fitted to offline data, drawing on set-based reachability analysis (Blanchini et al., 2008; Althoff et al., 2021; Girard, 2005). For the action-generating columns of G_k^{ctrl} and the center c_s (evaluated with zero initial position) of the fitted reachable set above, we compute the ratio as

$$\chi_\tau := \frac{\min_{\|v\|_2=1} \|(G_k^{\mathrm{ctrl}})^\top v\|_1}{\mathrm{Quan}_{s\sim\mathcal{D}} \|c_s\|_2},$$

Quantile is taken at 0.9, and we use $\chi_\tau \in (0.5, 1.5)$ as a heuristic range of comparable control and drift magnitudes. With this, we use $k = 1$ in `pointmaze` (zero drift), $k = 12$ in `antmaze` and $k = 48$ in `humanoidmaze`. These horizons are fixed across layouts and dataset types within each environment family before the critic training.

C.3 Training and Inference for Critic and Policy

Learning the ReQRL critic. To train our critic d_θ, we implement equation 13 using soft constraint penalties. Let $\Delta_\theta(s, g)$ denote the predicted descent on the left-hand side of equation 15. Our ReQRL objective is

$$\begin{aligned}\mathcal{L}_{ReQRL}(\theta) := & -\mathbb{E}_{s,g}\left[\phi(d_\theta(s,g))\right] \\ & + w_{\mathrm{in}}\,\mathbb{E}_{s,g}\left[(\Delta_\theta(s,g)-1)^2\right] \\ & + w_b\,\mathbb{E}_b\left[\mathrm{ReLU}\left(-\widehat{\nu}(x_s)^\top \nabla_{x_s} d_\theta(s,g)\right)^2\right],\end{aligned} \tag{16}$$

where $\phi(z)$ is the monotone distance transform from QRL (Wang et al., 2023, Eq. (40)). The parameters $w_{\mathrm{in}}, w_b > 0$ control the interior and boundary penalty weights, and $\mathbb{E}_b$ averages over sampled boundary (s, g) pairs. We follow EikHiQRL's (s, g) sampling for `antmaze` and `humanoidmaze` (Giammarino & Qureshi, 2026), while using random dataset goals for the `pointmaze` critic. Across all settings, the boundary penalty uses sampled states within the estimated boundary band. We use exact zonotopic support from equation 15, except in `pointmaze`, where we use a smooth log-sum-exp approximation and anneal its temperature from 1 to 0.05 during training, which converges to the exact support as the temperature tends to zero.

Policy training. We train policies by advantage-weighted regression (AWR) (Peng et al., 2019), using HIQL's simplified advantage estimates (Park et al., 2023). For $k = 1$, the policy $\pi_\psi(a \mid x_s, x_g)$ predicts primitive actions and is trained on recorded transitions:

$$\begin{aligned}A(s, s', g) &= d_\theta(s, g) - d_\theta(s', g), \\ \mathcal{L}_\pi(\psi) &= -\mathbb{E}_{s,a,s',g}\left[\min\{\exp(\beta A(s, s', g)), C\} \log \pi_\psi(a \mid x_s, x_g)\right].\end{aligned}$$

For $k > 1$, $\pi_\psi(w \mid x_s, x_g)$ predicts waypoints through offsets from x_s. Its targets are k-step hindsight waypoints $w = x_{s_{t+k}}$, capped at the trajectory end or the sampled future goal's time, whichever comes first:

$$\begin{aligned}A^h(s, w, g) &= d_\theta(x_s, x_g) - d_\theta(w, x_g), \\ \mathcal{L}_{\pi_h}(\psi) &= -\mathbb{E}_{s,w,g}\left[\min\{\exp(\beta_h A^h(s, w, g)), C\} \log \pi_\psi(w \mid x_s, x_g)\right].\end{aligned}$$

A separate low-level value function takes the full state s and coordinate waypoint w. Its two heads use action-free, goal-conditioned expectile TD learning with Polyak target networks, following HIQL and EikHiQRL (Kostrikov et al., 2021; Park et al., 2023; Giammarino & Qureshi, 2026), without a learned waypoint embedding or an additional Eikonal penalty on the low-level value. Writing V_η for the mean of these heads, we train the action policy on recorded actions and k-step waypoints truncated at the trajectory end:

$$\begin{aligned}A^\ell(s, s', w) &= V_\eta(s', w) - V_\eta(s, w), \\ \mathcal{L}_{\pi_\ell} &= -\mathbb{E}_{s,a,s',w}\left[\min\{\exp(\beta_\ell A^\ell(s, s', w)), C\} \log \pi_\ell(a \mid s, w)\right].\end{aligned}$$

All advantage weights are detached during policy updates, with $C = 100$. Following EikHiQRL, policy goals are uniformly sampled future states for `navigate`, an equal mixture of future and random dataset states for `stitch`, and random dataset states for `explore`. Low-level value training uses the critic-goal sampling described above. These policy-training components retain the corresponding baseline settings; we use coordinate waypoints directly and omit the additional Eikonal penalty on the low-level value function.

Algorithm 1 ReQRL policy execution

Require: Initial state $s_0 \in \mathcal{S}$, goal $g \in \mathcal{S}$, horizon k, episode limit $T_{\max}$
Require: Trained actor π_ψ for ReQRL critic d_θ and, for $k > 1$, low-level actor π_ℓ
Extract goal coordinates x_g
for $t = 0, \ldots, T_{\max} - 1$ **do**
  Extract x_{s_t} from s_t
  if $k = 1$ **then**
    $a_t \leftarrow \mathrm{clip}(\mu_\psi(x_{s_t}, x_g), -1, 1)$ ▷ Mean primitive action
  else
    $w_t \leftarrow x_{s_t} + \mathrm{clip}(\mu_\psi(x_{s_t}, x_g), -10, 10)$ ▷ Mean offset to waypoint
    $a_t \leftarrow \mathrm{clip}(\mu_\ell(s_t, w_t), -1, 1)$ ▷ Waypoint-conditioned action
  end if
  Execute a_t and observe s_{t+1}
  if the goal is reached or the episode terminates or is truncated **then**
    break
  end if
end for

Inference. For $k = 1$, the trained policy $\pi_\psi(a \mid x_s, x_g)$ directly predicts primitive actions. For $k > 1$, the waypoint policy $\pi_\psi(w \mid x_s, x_g)$ predicts an offset from x_s, and the low-level policy $\pi_\ell(a \mid s, w)$ maps the full observation and resulting waypoint to an action. We use mean outputs, denoted by μ_ψ and μ_ℓ, with componentwise clipping. Waypoints are recomputed at every environment step. Evaluation ends upon goal achievement, environment termination, or the benchmark's episode time limit. Refer to Algorithm 1.

C.4 Baselines and their Implementation

We compare against quasimetric methods (QRL, EikHiQRL) and non-quasimetric methods (GCIQL, CRL, HIQL, SAW, OTA, GAS). GCIQL (Kostrikov et al., 2021) learns a goal-conditioned value by expectile regression. HIQL (Park et al., 2023) learns an action-free value and extracts subgoal and low-level policies from it. SAW (Zhou & Kao, 2026) trains a single flat policy whose actions for distant goals are regressed toward those of a short-horizon policy. CRL (Eysenbach et al., 2022) trains the critic as a contrastive classifier over future states. OTA (Ahn et al., 2026) learns the high-level value over temporally abstracted transitions. GAS (Baek et al., 2025) plans over a graph of key states in a learned temporal-distance space. Among the qausimetric methods, QRL (Wang et al., 2023) learns an IQE critic with transition-cost constraints and EikHiQRL (Giammarino & Qureshi, 2026) adds an Eikonal gradient penalty and a hierarchical actor of HIQL.

All methods are reimplemented in `PyTorch` that follow each method's released `JAX` code, sharing the same training loop, evaluation protocol, and seeds. All baselines including our method are written in `PyTorch` and are compiled before running. Hyperparameters follow the authors' per-environment settings when available and are otherwise tuned, all following the same sweep protocol.

C.5 Training and Hyperparameters

All networks use three hidden layers of width 512 with GELU activations. Critics additionally use LayerNorm. The quasimetric critic uses IQE (Wang & Isola, 2022) with a 512-dimensional latent representation, organized into 64 components of size 8. Our QRL and EikHiQRL baselines use the same IQE critic architecture. Actors parameterize diagonal Gaussian distributions with fixed unit standard deviation, and the low-level value function has two heads. We optimize the networks with `Adam` and batch size 1024. The inherited policy-training settings follow EikHiQRL: the low-level discount is $\gamma = 0.995$, the expectile is 0.7, and the Polyak update coefficient is 0.005. AWR

coefficients are $\alpha = 3$, except for `antmaze-explore`, which uses $\alpha = 10$. Our additional hyperparameters are ReQRL's critic loss weights, w_{in} and w_b. the resulting values of w_{in} and w_B are in Table 5. All methods receive the same number of training updates within each setting: 10^5 training steps for `pointmaze` and `antmaze`, except 10^6 steps for `antmaze-explore`, and 10^6 training steps for `humanoind` and `antsoccer`.

Environment	Dataset	w_{in}	w_b
`pointmaze`	`navigate`	$(0.5, 1, 1) \times 10^2$	$(1, 1, 5) \times 10^1$
	`stitch`	$(2.5, 2, 2) \times 10^2$	$(0.5, 1, 1.5) \times 10^1$
`antmaze`	`navigate`	$(3, 3, 3)$	$(1, 1, 3) \times 10^2$
	`stitch`	$(2.5, 2.5, 5) \times 10^{-1}$	$(1, 1, 0.5) \times 10^2$
	`explore`	$(2.5, 2.5) \times 10^{-1}$	$(1, 0.5) \times 10^2$
`humanoidmaze`	`navigate`	$(5, 5, 5)$	$(3, 3, 5) \times 10^2$
	`stitch`	$(0.5, 0.5, 1)$	$(1, 1, 1) \times 10^2$

Table 5: Selected penalty weights w_{in} and w_b, ordered as (`medium`, `large`, `giant`). For `explore`, entries correspond to (`medium`, `large`).

D Additional Experiments Details and Results

We include additional experimental results and details in this section. Specifically, we provide the full result tables supplementing the main text, experimental details and additional results for `carmaze` and `antsoccer`, and runtime comparison between ReQRL and our reimplemented `PyTorch` baselines.

Env	Data	Size	Other GCRL						Quasimetric		
			CRL	SAW	GCIQL	HIQL	OTA	GAS	QRL	EikHiQRL	**ReQRL**
carmaze	nav	medium	50 ± 5	52 ± 7	28 ± 4	53 ± 5	63 ± 7	81 ± 5	36 ± 6	72 ± 8	$\mathbf{\underline{90 \pm 3}}$
		large	13 ± 4	2 ± 2	4 ± 1	19 ± 8	32 ± 14	67 ± 7	4 ± 4	45 ± 15	$\mathbf{\underline{81 \pm 12}}$
		giant	0 ± 0	0 ± 0	0 ± 0	1 ± 1	3 ± 3	40 ± 8	1 ± 1	13 ± 9	$\mathbf{\underline{60 \pm 14}}$
	sti	medium	28 ± 5	27 ± 11	10 ± 2	23 ± 7	60 ± 6	74 ± 6	30 ± 6	69 ± 9	$\mathbf{\underline{82 \pm 6}}$
		large	0 ± 0	1 ± 1	0 ± 1	0 ± 1	14 ± 6	64 ± 10	1 ± 1	26 ± 9	$\mathbf{\underline{78 \pm 12}}$
		giant	0 ± 0	0 ± 0	0 ± 0	0 ± 0	0 ± 0	31 ± 8	0 ± 0	5 ± 7	$\mathbf{\underline{45 \pm 17}}$

Table 6: **Success rates (%)** on `carmaze` at the final checkpoint (mean $\pm$ std over 10 seeds; 50 episodes per goal). `nav` and `sti` denote `navigate` and `stitch`. Means at least 95% of the best are **bold**; the best is underlined.

Additional details on `carmaze`. We elaborate on the `carmaze` environment and datasets discussed and evaluated in Section 6 and Table 6. We integrate the `Safety-Gymnasium Car` robot (Ji et al., 2023), simulated in `MuJoCo` (Todorov et al., 2012), into `OGBench`'s official `medium`, `large`, and `giant` layouts (Park et al., 2025). The 22-D state space $\mathcal{S}$ consists of the 3-D chassis position, two 4-D orientation quaternions for the chassis and caster, and 11 generalized velocities. The two unbounded wheel angles are omitted, while their angular velocities are retained. Its 2-D actions are in $\mathcal{A} := [-1, 1]^2$ and control the two wheel motors with torque scale 0.02. We set wheel damping to 0.0005 and use a 0.1 s control interval, comprising 25 simulator steps of 0.004 s. Following `OGBench`'s collection pipeline, we train an SAC locomotion expert for 400k steps, then generate trajectories using waypoint guidance with Gaussian action noise. Each `navigate` dataset contains 1,000 trajectories, with approximately 0.6M, 1M, and 1.6M samples for `medium`, `large`, and `giant`, respectively. Each `stitch` dataset contains 5,000 short trajectories, totaling approximately 1M samples per layout.

For estimating the reachable set and estimate boundary normals, ReQRL uses critic goal-coordinates $x_s = (x, y) \in \mathbb{R}^2$ and reduced states $r_s = (x, y, v_x, v_y, \cos\psi, \sin\psi) \in \mathbb{R}^6$, as for `antmaze` (see Appendix C). We also fix $k = 16$ across all layouts and dataset types, consistent with the horizon-selection rationale in Appendix C.2. The remaining architecture, estimation, and policy-training details follow Appendix C. Training uses 10^5 training updates; GAS uses 10^5 updates in each of

Env	Data	Size	Other GCRL						Quasimetric		
			CRL	SAW	GCIQL	HIQL	OTA	GAS	QRL	EikHiQRL	**ReQRL**
pointmaze	nav	medium	34 ± 5	1 ± 1	56 ± 2	3 ± 2	2 ± 1	1 ± 1	33 ± 7	2 ± 4	7 ± 6
		large	49 ± 9	27 ± 12	82 ± 2	32 ± 7	6 ± 2	0 ± 0	40 ± 7	0 ± 1	5 ± 4
		giant	58 ± 8	50 ± 13	93 ± 7	59 ± 12	14 ± 5	2 ± 2	43 ± 9	9 ± 1	10 ± 7
	sti	medium	72 ± 12	31 ± 8	78 ± 3	33 ± 8	13 ± 3	6 ± 6	32 ± 7	2 ± 4	9 ± 7
		large	65 ± 13	55 ± 8	71 ± 3	68 ± 10	20 ± 10	3 ± 5	37 ± 4	1 ± 1	7 ± 5
		giant	72 ± 12	57 ± 15	81 ± 8	76 ± 10	57 ± 14	0 ± 0	47 ± 12	14 ± 15	22 ± 8
antmaze	nav	medium	8 ± 1	4 ± 1	11 ± 1	4 ± 1	3 ± 0	2 ± 0	5 ± 1	3 ± 0	1 ± 0
		large	9 ± 1	7 ± 1	11 ± 1	6 ± 1	3 ± 0	3 ± 1	7 ± 2	5 ± 0	2 ± 0
		giant	14 ± 1	15 ± 1	14 ± 1	14 ± 1	9 ± 1	8 ± 3	14 ± 1	6 ± 1	4 ± 1
	sti	medium	15 ± 1	9 ± 1	12 ± 1	9 ± 1	3 ± 0	2 ± 0	6 ± 1	4 ± 0	1 ± 0
		large	17 ± 1	15 ± 1	11 ± 1	15 ± 1	8 ± 1	3 ± 1	10 ± 1	5 ± 1	1 ± 0
		giant	16 ± 2	19 ± 1	13 ± 1	15 ± 1	13 ± 1	5 ± 1	16 ± 1	7 ± 1	4 ± 1
	exp	medium	8 ± 3	17 ± 2	23 ± 1	15 ± 2	2 ± 1	4 ± 1	21 ± 2	13 ± 3	1 ± 1
		large	6 ± 2	18 ± 1	21 ± 1	17 ± 2	5 ± 2	5 ± 1	19 ± 1	11 ± 2	1 ± 1
humanoid	nav	medium	21 ± 1	10 ± 1	25 ± 1	9 ± 1	8 ± 1	7 ± 2	16 ± 4	9 ± 1	3 ± 0
		large	23 ± 2	13 ± 1	24 ± 1	15 ± 2	9 ± 1	9 ± 1	18 ± 3	13 ± 2	10 ± 3
		giant	33 ± 4	17 ± 2	28 ± 2	23 ± 2	9 ± 1	11 ± 1	25 ± 2	11 ± 1	7 ± 1
	sti	medium	22 ± 1	9 ± 1	21 ± 1	8 ± 1	7 ± 0	6 ± 1	15 ± 2	10 ± 1	4 ± 1
		large	26 ± 2	12 ± 1	23 ± 2	13 ± 1	10 ± 1	9 ± 1	18 ± 2	13 ± 1	6 ± 0
		giant	32 ± 3	25 ± 2	26 ± 2	26 ± 2	12 ± 1	13 ± 1	27 ± 2	13 ± 1	7 ± 1

Table 7: **Collision rate (%)** (↓): percentage of time steps with agent-boundary contact in MuJoCo (mean ± std over 10 seeds). nav, sti, exp are shorthands for navigate, stitch, explore datasets of OGBench, respectively.

Env	Size	Other GCRL		Quasimetric		
		CRL	HIQL	QRL	EikHiQRL	**ReQRL**
antsoccer-nav	arena	24 ± 2	$\mathbf{60 \pm 4}$	10 ± 3	$\mathbf{61 \pm 5}$	$\underline{\mathbf{65 \pm 3}}$
	medium	4 ± 2	13 ± 3	2 ± 2	13 ± 1	$\underline{\mathbf{17 \pm 3}}$

Table 8: **Success rates (%)** on OGBench's antsoccer-navigate after 10^6 training updates (mean ± std). For reference, baseline results are taken from Giammarino & Qureshi (2026) under the same training budget.

its temporal-distance and policy-training stages, with graph construction between them. We tune method-specific hyperparameters for ReQRL and all baselines with equal effort. We initialize experiments with these selected configurations, additionally sweeping QRL, HIQL, OTA, and EikHiQRL and conducting a local search for GAS. For ReQRL specifically, we sweep $w_{\text{in}} \in \{0.25, 1, 5, 100\}$ and $w_b \in \{1, 10, 100\}$ and use $(w_{\text{in}}, w_b) = (5, 100)$ for all navigate layouts, $(1, 10)$ for medium/large-stitch, and $(0.25, 1)$ for giant-stitch. Final evaluation follows Appendix C. The full comparison in Table 6 supports the main-text results, with ReQRL achieving the highest mean success in all six settings.

Supplementary experimental results. In Table 7, the full collision results are consistent with the main-text comparison on antmaze and humanoidmaze. ReQRL achieves lower mean collision rates than QRL and EikHiQRL across settings, and generally lower or comparable rates relative to OTA and GAS. In pointmaze, however, the point mass can touch corners or slide along walls while progressing toward the goal, so successful trajectories can still accumulate contact steps, which is consistent with the reported results in Table 1.

We additionally evaluate ReQRL on OGBench's antsoccer tasks (Park et al., 2025), where goal reaching requires moving an external object through contact. An ant must dribble a ball toward a target in the open arena or the medium maze. The 42-D state combines the 29-D ant state and 13-D ball state, with eight joint-torque actions in $\mathcal{A} = [-1, 1]^8$. We use 10^6 training updates with $k = 24$, otherwise following Appendix C. The critic coordinates $x_s \in \mathbb{R}^4$ contain the planar positions of the ant and ball. The reduced state $r_s \in \mathbb{R}^{12}$ contains the ant's planar position, planar velocity,

and heading sine/cosine, together with the ball's 3-D position and linear velocity. Following the benchmark's goal observations, both goal-position blocks use the target location, while evaluation success depends only on the ball reaching within 0.5 m of the target. As in Appendix C.2, we fit the affine dynamics model, projecting the propagated reachable sets onto x_s. Table 8 reports mean success rates of 65% and 17%, compared with 61% and 13% for EikHiQRL using the published, best-of-10 evaluations at 1M (Giammarino & Qureshi, 2026). These results extend our evaluation to goal-reaching tasks involving object interaction.

In Figure 7, we also compare success rate against wall-clock training time on the `humanoidmaze` environment, using our compiled `PyTorch` implementations on a single `A100` GPU for 10^6 training updates. Success rates are averaged across the `medium`, `large`, and `giant` layouts, separately for `navigate` and `stitch`. Overall, these results complement the main-text comparisons by showing competitive success together with practical training efficiency.

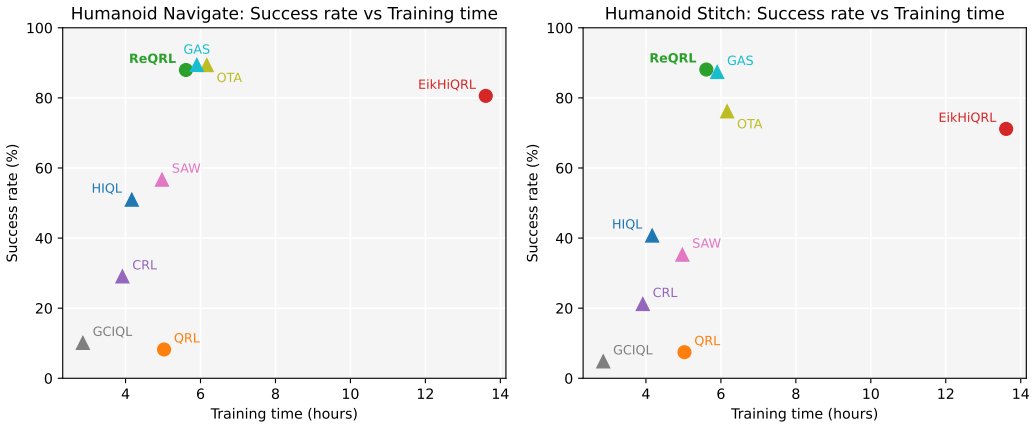

Figure 7: Success rate averaged across maze sizes versus training time on `humanoidmaze navigate` and `stitch`. Wall-clock time covers 10^6 training updates on one `A100` using compiled `PyTorch` implementations. Circles denote quasimetric methods; triangles denote other GCRL baselines.