2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Quasimetric Decision Transformers: Enhancing Goal-Conditioned Reinforcement Learning with Structured Distance Guidance

Madhav Goyani[1], Heidar Davoudi[1], and Mehran Ebrahimi[1]

***Abstract*— Recent works have shown that tackling offline reinforcement learning (RL) with a conditional policy produces promising results. Decision Transformers (DT) have shown promising results in offline reinforcement learning by leveraging sequence modeling. However, standard DT methods rely on return-to-go (RTG) tokens, which are heuristically defined and often suboptimal for goal-conditioned tasks. In this work, we introduce Quasimetric Decision Transformer (QuaD), a novel approach that replaces RTG with learned *quasimetric distances*, providing a more structured and theoretically grounded guidance signal for long-horizon decision-making. We explore two quasimetric formulations: *interval quasimetric embeddings (IQE)* and *metric residual networks (MRN)*, and integrate them into DTs. Extensive evaluations on the *AntMaze benchmark* demonstrate that QuaD outperforms standard Decision Transformers, achieving state-of-the-art success rates and improved generalization to unseen goals. Our results suggest that quasimetric guidance is a viable alternative to RTG, opening new directions for learning structured distance representations in offline RL.**

I. Introduction

Reinforcement Learning (RL) has achieved remarkable success in domains such as robotics, autonomous driving [1], and game playing [2], by enabling agents to learn optimal policies through trial-and-error interaction with an environment. However, these successes typically rely on online RL paradigms, which are often impractical in real-world settings due to sample inefficiency, safety concerns, and computational constraints.

Offline RL addresses these challenges by learning from static datasets of prior interactions without further environment access [3]. It enables safe and efficient learning but suffers from the challenge of distributional shift, where the learned policy encounters out-of-distribution states or actions during inference [4]. A recent family of models, known as Decision Transformers (DTs) [5], has approached RL from a sequence modeling perspective, treating trajectories as sequences to be predicted via autoregressive transformers. DTs condition action prediction on past states, actions, and a Return-to-Go (RTG) token, representing the desired cumulative future reward.

While DTs have shown promise, their reliance on RTG presents critical limitations in goal-conditioned RL (GCRL) environments. In these settings, tasks are framed as reaching a specific goal state, and rewards are often sparse or binary. Consequently, RTG becomes an arbitrary and uninformative signal during most of the trajectory, particularly in long-horizon tasks such as AntMaze [6], where DTs perform poorly in medium and large maze configurations. Furthermore, the use of naïve mean squared error (MSE) loss functions treats all actions equally, failing to emphasize high-value behaviors required for successful goal completion.

This paper introduces the **Quasimetric Decision Transformer (QuaD)**, a novel approach that replaces RTG conditioning with a learned quasimetric function $d(s, g)$, which estimates the directional difficulty of reaching a goal state g from a current state s. Unlike scalar reward aggregates, quasimetric functions provide structured and continuous guidance that aligns more naturally with goal-directed behavior. We explore two such formulations: Interval Quasimetric Embedding (IQE) [7] and Metric Residual Networks (MRN) [8], each capturing asymmetric transition difficulty in high-dimensional spaces.

In addition, we augment the DT training objective with value-aware loss functions, including Advantage-Weighted Regression (AWR) [9] and DDPG with Behavior Cloning (DDPG+BC) [10], to prioritize high-value actions and address the limitations of MSE. Through extensive experiments on the AntMaze benchmark, we demonstrate that QuaD significantly outperforms baseline DTs, behavior cloning, and value-based methods, particularly in sparse-reward and long-horizon tasks.

In summary, our key contributions are:

- We propose replacing RTG in Decision Transformers with a learned quasimetric signal that better reflects goal-reaching difficulty.
- We introduce value-aware objectives to move beyond imitation learning and enhance goal-directed behavior.
- We evaluate two quasimetric architectures, IQE and MRN, to structure the goal-space representation.
- We conduct experiments showing that QuaD outperforms strong offline RL baselines on challenging AntMaze tasks.

Our results indicate that structured distance signals and value-guided optimization are essential to bridging the gap between sequence modeling and effective goal-conditioned RL.

II. Related Work

Our work builds on previous work in learning temporal distances, concepts from goal-conditioned RL and sequential modeling for reinforcement learning. Our analysis will draw a connection between these prior methods, a connection which will ultimately result in a new guiding metric for decision transformer for goal-conditioned environments.

[1]Madhav Goyani, Heidar Davoudi, and Mehran Ebrahimi are with the Faculty of Science, Ontario Tech University, 2000 Simcoe St N, Oshawa, ON L1G 0C5, Canada madhav.goyani@ontariotechu.net

A. Goal Conditioned Reinforcement Learning

Goal-conditioned reinforcement learning (GCRL) provides a flexible framework for training policies to achieve diverse outcomes by conditioning on explicit goal states. Unlike traditional reinforcement learning (RL), which optimizes for cumulative rewards, GCRL shifts the focus toward reaching specific states in the environment, making it particularly useful for tasks where defining a dense reward function is challenging or infeasible. A key challenge in GCRL is learning effective goal-conditioned value functions. Several approaches leverage hindsight relabeling [11], contrastive learning [12], and state-occupancy matching to improve generalization and robustness. However, many of these methods rely on bootstrapping with a learned value function, which can introduce instability and inefficiencies, particularly in long-horizon tasks with sparse rewards [13]. To mitigate the challenges of long-horizon planning, hierarchical RL (HRL) [14] and subgoal planning [15] have been explored as extensions to GCRL. HRL methods decompose tasks into subgoals and learn policies that operate at multiple temporal resolutions, improving sample efficiency and task scalability.

B. Transformers for Reinforcement Learning

Transformers have shown remarkable generalization capabilities in fields such as language modeling, image generation, and representation learning [16], [17], [18]. Within offline RL, transformer-based policies treat RL tasks as sequential prediction problems. Decision Transformer [5] models trajectories as sequences and autoregressively predicts actions conditioned on return-to-go, past states, and actions. The Trajectory Transformer [19] demonstrates transformer-based learning for single-task offline policies. Multi-game Decision Transformer [20] and Gato [21] extend transformer-based policies to multi-task and cross-domain applications. However, these approaches distill expert policies rather than enabling self-improvement. When data are suboptimal or adaptation to new tasks is required, multi-game DTs must fine-tune parameters, and Gato must rely on expert demonstrations. If the model generalizes effectively to out-of-distribution return-to-go values, it can generate superior policies by prompting higher returns. However, achieving this level of generalization remains an open challenge in sequential decision-making. DT struggles with robustness to data distribution shifts, particularly when trained on trajectories generated by suboptimal policies. Research indicates that DT underperforms in tasks requiring trajectory stitching, integrating suboptimal trajectory segments to create improved policies[22], [23], [24]. This confirms that naive return-to-go prompts are insufficient for solving complex sequential decision-making problems.

C. Metric Learning in RL and State Abstractions for Decision Making

A fundamental challenge in reinforcement learning (RL) is learning representations that capture meaningful distances between states. Successor representations and successor features [25], [26] offer one approach by using temporal difference learning to predict states visited in the future. While these methods bear similarity to Q-learning [27] in tabular settings, they struggle with continuous states and actions [19], [28]. To address this, recent work [12], [28]has proposed learning representations where inner products correspond to visitation probabilities. The notion of state-space geometry plays a key role in RL. Prior work has explored quasimetrics for multi-task planning [29] and parametrizing Q-functions with improved goal-reaching performance in DDPG [10] and HER [11]). Other approaches define distances based on optimal value functions, the Wasserstein-1 distance [30], or bisimulation metrics [31], [32]. A key advantage of quasimetrics is their ability to capture transition difficulty between states while satisfying the triangle inequality. We utilize a quasimetric that can be easily learned from discounted state occupancy measures, providing a principled way to model goal-conditioned value functions without assuming symmetry or other restrictive properties. By leveraging state abstraction techniques and quasimetric learning, our approach enables improved long-horizon generalization and more effective goal-reaching policies.

D. Offline Policy Optimization: AWR vs. DDPG+BC

Recent advances in offline reinforcement learning have explored hybrid methods that combine value-based learning with supervised behavioral cloning. Two widely studied techniques in this space are **Advantage-Weighted Regression (AWR)** and **DDPG with Behavior Cloning (DDPG+BC)**. AWR [9] is a policy optimization technique that estimates advantages using a fixed critic and then performs a weighted regression to update the policy. Unlike traditional actor-critic methods that rely on gradient-based policy updates, AWR performs non-parametric advantage-weighted behavioral cloning. This yields a stable policy improvement method well-suited to offline data, where overestimation of values can be harmful. AWR introduces a temperature hyperparameter that controls the tradeoff between policy entropy and exploitation of the learned critic. DDPG+BC [22] is a modification of the Deep Deterministic Policy Gradient (DDPG) algorithm [10] that incorporates a behavioral cloning regularization term in the policy loss. This term encourages the learned policy to remain close to the behavior policy observed in the offline dataset, stabilizing learning and mitigating value overestimation. Unlike AWR, DDPG+BC relies on backpropagation through both actor and critic networks and supports deterministic policy updates. In our work, we explore both AWR and DDPG+BC losses within the QuaD framework as alternative optimization objectives for guiding the transformer's action predictions. This allows us to study how different forms of value-aware imitation affect policy learning under quasimetric supervision.

III. Preliminaries

In this section, we introduce notation and preliminary definitions for goal-conditioned RL, the Decision Transformer [5] method and the notion of quasimetrics [7], [33], [8] which will serve as the foundation for this work.

A. Problem Setting

The offline goal-conditioned reinforcement learning (GCRL) problem is defined by a controlled Markov process $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mu, p)$ (a Markov decision process (MDP) without rewards) along with an unlabeled dataset $\mathcal{D}$. Here, $\mathcal{S}$ denotes the state space, $\mathcal{A}$ represents the action space, $\mu(s) \in \Delta(\mathcal{S})$ is the initial state distribution, and $p(s' \mid s, a) : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ describes the transition dynamics. The notation $\Delta(\mathcal{X})$ refers to the space of probability distributions over a set $\mathcal{X}$. The dataset $\mathcal{D} = \{\tau^{(n)}\}_{n=1}^{N}$ consists of N unlabeled trajectories:

$$\tau^{(n)} = (s_0^{(n)}, a_0^{(n)}, r_0^{(n)}, s_1^{(n)}, a_1^{(n)}, r_1^{(n)}, \ldots, s_T^{(n)}, a_T^{(n)}, r_T^{(n)}).$$

The objective of offline GCRL is to learn a goal-conditioned policy $\pi(a \mid s, g) : \mathcal{S} \times \mathcal{S} \to \Delta(\mathcal{A})$ that enables an agent to reach any target state $g \in \mathcal{S}$ from any initial state in the minimum number of time steps. This is achieved by maximizing the expected return:

$$\mathbb{E}_{\tau \sim p(\tau|g)} \left[\sum_{t=0}^{T} \gamma^t \delta_g(s_t) \right], \tag{1}$$

where $T \in \mathbb{N}$ is the episode horizon, $\gamma \in (0, 1)$ is the discount factor, and $p(\tau \mid g)$ is the trajectory distribution induced by:

$$p(\tau \mid g) = \mu(s_0) \prod_{t=0}^{T-1} \pi(a_t \mid s_t, g) p(s_{t+1} \mid s_t, a_t).$$

Here, $\delta_g(s)$ represents the Dirac delta function, which in a discrete MDP corresponds to the indicator function $\mathbf{1}_{\{g\}}(s)$. In continuous MDPs, a precise definition requires measure-theoretic notation or distribution theory, but we omit these details for simplicity.

For any goal $g \in \mathcal{S}$, we frame goal-reaching as an inference problem [34], [35], [36], [12]: given the current state and desired goal, what is the most likely action to bring the agent closer to that goal? This corresponds to solving the MDP $\mathcal{M}_g$, which extends $\mathcal{M}$ with a goal-conditioned reward function:

$$r_g(s) = (1 - \gamma)\delta_g(s). \tag{2}$$

Thus, a goal-conditioned policy $\pi(a \mid s, g)$ receives both the current state and goal as inputs, effectively transforming $\mathcal{M}$ into a goal-conditioned MDP, denoted as $\mathcal{M}_g$.

B. Revisiting Decision Transformers

Decision Transformer (DT) [5] is an influential method that bridges sequence modeling with decision-making by adapting the transformer architecture [16] to reinforcement learning. Unlike traditional reinforcement learning (RL) algorithms that rely on dynamic programming or policy gradient methods, DT directly learns an autoregressive model from trajectory data using a causal transformer [37]. This allows DT to leverage powerful pre-trained architectures developed for language and vision tasks [38], [39]. DT modifies initial trajectories from the dataset and represents them as :

$$\tau = (R_1, s_1, a_1, R_2, s_2, a_2, \ldots, R_T, s_T, a_T), \tag{3}$$

where $R_t = \sum_{i=t}^{T} r_i$ is the return-to-go (RTG) from time step t onward. The DT policy is parameterized as:

$$\pi_{\text{DT}}(a_t | s_t, R_t, \tau_t), \tag{4}$$

where $\tau_t = (R_0, s_0, a_0, \ldots, R_{t-1}, s_{t-1}, a_{t-1})$ is the sub-trajectory history before time step t. Training is performed autoregressively, where the model predicts actions conditioned on the previous state, RTG, and trajectory history. At test time, DT initializes with a desired return-to-go R_0 and an initial state s_0. The generated action is executed, the return is decremented by the achieved reward, and the process continues until termination. The authors of [5] argue that the conditional prediction model is able to perform policy optimization without using dynamic programming. However, recent works observe that DT often shows inferior performance compared to dynamic programming based offline RL algorithms when the offline dataset consists of sub-optimal trajectories [22], [23], [24].

C. Learning the Quasimetric Distance Function

Within a Markov decision process (MDP), one can naturally think of a notion of "distance" between states that reflects how challenging it is to transition from one state to another. Intuitively, several quantities could serve this purpose: the probability of reaching a target state within a fixed horizon, the expected time required to reach it, or the probability of ever reaching it under a given policy. For such a notion of distance to be meaningful in goal-reaching settings, it should satisfy a structural consistency property namely, the triangle inequality,

$$d(a, c) \leq d(a, b) + d(b, c).$$

This expresses the idea that if it is feasible to move from a to b and from b to c, then transitioning directly from a to c should not be more difficult than completing the two intermediate steps. When this function is symmetric and satisfies the standard axioms, it defines a *metric*; in more general (possibly asymmetric) cases, it is referred to as a *quasimetric* [33], [7], [8], [40], [**?**].

Definition 1: We define a distance function $d : S \times S \to \mathbb{R}$ that satisfies nonnegativity and identity properties. The set of all such distance functions is given by:

$$\mathcal{D} \triangleq \{d : S \times S \to \mathbb{R} \mid d(s, s) = 0,\ d(s, s') > 0 \text{ for all } s \neq s' \in S\}. \tag{5}$$

A distance function satisfying the triangle inequality is called a quasimetric, and the set of all quasimetrics is:

$$\mathcal{Q} \triangleq \{d \in \mathcal{D} \mid d(s, g) \leq d(s, w) + d(w, g) \text{ for all } s, g, w \in S\}. \tag{6}$$

Previous approaches, such as bisimulation-based methods [31], construct such distances using the reward function as a guiding signal. In contrast, our objective is to learn a distance representation that does not depend explicitly on reward design. With an appropriate formulation, learning a

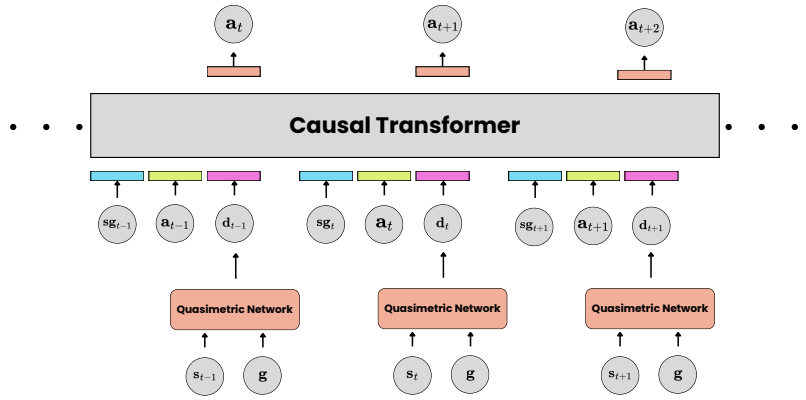

Fig. 1: **Architecture of the Quasimetric Decision Transformer (QuaD)**. The model replaces return-to-go (RTG) with a learned quasimetric function $d(s_t, g)$, which provides structured goal-aware guidance. The Quasimetric Network computes d_t given the current state s_t and the goal g, producing a distance embedding. These embeddings, along with state-goal embeddings sg_t and past actions a_t, are tokenized and processed by a causal transformer, which autoregressively predicts actions a_{t+1}. The quasimetric function enables better trajectory modeling and generalization in goal-conditioned RL tasks.

goal-conditioned value function can be viewed as identifying a distance structure that facilitates effective goal reaching. Architectures that explicitly encode metric constraints—such as Metric Residual Networks (MRN) [8] and Interval Quasimetric Embeddings (IQE) [33], [7]—provide practical mechanisms for learning such structured representations.

IV. Quasimetric Guided Decision Transformer

The Quasimetric Decision Transformer (QuaD) replaces RTG with a learned quasimetric function $d(s, g)$, which explicitly models the difficulty of reaching a goal state g from a given state s. This quasimetric satisfies the properties discussed in Section 3.3 and provides a structured distance measure for goal-reaching tasks.

A QuaD trajectory is represented as:

$$\tau = (s_1, a_1, d(s_1, g), s_2, a_2, d(s_2, g), \dots), \tag{7}$$

where $d(s_t, g)$ replaces the return-to-go R_t.

The core idea behind QuaD is that $d(s, g)$ acts as a ***structured guidance signal***, allowing the transformer model to (1) learn more effective trajectory stitching by minimizing $d(s, g)$ at each step, (2) Generalize to new goals based on quasimetric-based similarity in state space.

A. Quasimetric Models in Goal-Conditioned MDPs

A quasimetric model d_θ usually consists of (1) a deep encoder mapping inputs in $\mathcal{X}$ to a generic latent space $\mathbb{R}^d$ and (2) a differentiable latent quasimetric head $d_{\mathsf{latent}} \in (\mathbb{R}^d)$ that computes the quasimetric distance for two input latents. θ contains both the parameters of the encoder and parameters of the latent head d_{latent}, if any. Recent works have proposed many choices of d_{latent}, which have different properties and performances. We refer interested readers to [33] for an in-depth treatment of such models. The quasimetric model d_θ is optimized as follows:

$$\begin{aligned} &\max_{\theta} \ \mathbb{E}_{\substack{s\sim p_{\text{state}} \\ g\sim p_{\text{goal}}}}[d_\theta(s, g)] \\ &\text{subject to } \mathbb{E}_{(s,a,s',r)\sim p_{\text{transition}}}[\texttt{relu}(d_\theta(s, s') + r)^2] \le \epsilon^2, \end{aligned} \tag{8}$$

where $\varepsilon > 0$ is small, and $\texttt{relu}(x)$ prevents $d_\theta(s, s')$ from exceeding the transition cost $-r \ge 0$. After optimization, we take d_θ as our estimate of the difficulty of reaching a goal state g from a given state s.

1) Training QuaD with Quasimetric Distance: Using mean-squared-error loss alone in Decision Transformer (DT) can lead to suboptimal policy learning, as it directly minimizes the difference between predicted and observed actions without considering long-term rewards. This approach lacks a mechanism to distinguish high-value actions from suboptimal ones, limiting performance in offline RL settings. To address this, we integrate Deep Deterministic Policy Gradient with Behavior Cloning (DDPG+BC) [10], combining Q-function optimization with policy regularization. The QuaD training objective follows the standard Decision Transformer loss function but conditions on the quasimetric distance $d(s, g)$:

$$\begin{aligned} \mathcal{L}_{\text{QuaD}} = \lambda \cdot \mathbb{E}_{(s,a)\sim\mathcal{D}}\left[-Q(s, g)\right] \\ + (1-\lambda) \cdot \mathbb{E}_{(s,a)\sim\mathcal{D}}\left[\|\hat{a} - a\|^2\right], \end{aligned} \tag{9}$$

The first term, $\mathbb{E}[-Q(s, g)]$, promotes actions that yield higher Q-values. Minimizing the negative Q-value encourages value-driven behavior. The second term, $\mathbb{E}[\|\hat{a} - a\|^2]$, is a standard mean squared error loss between the predicted action $\hat{a}$ and the ground-truth action a from the dataset, encouraging imitation of demonstrated behavior.

DDPG provides value-based updates, ensuring the policy prioritizes high-reward actions, while BC prevents excessive deviation from the dataset, improving stability. The additional MSE loss refines action consistency, keeping predictions

Fig. 2: D4rl AntMaze environments - Umaze, Medium & Large

aligned with observed behaviors while benefiting from value-driven learning. Furthermore, instead of treating goals and states as separate tokens as done by DT, we enhance trajectory tokenization by concatenating the goals with state together and then tokenize the vector, improving context understanding. This integrated approach results in better stability, improved action selection, and more effective offline RL training

The quasimetric function $d(s, g)$ is learned separately as a neural network $f_\theta(s, g)$ trained to satisfy the quasimetric properties:

$$d(s, g) \approx \min_{\pi} \mathbb{E}_\pi \left[\sum_{t=0}^{T} c(s_t, g) \mid s_0 = s \right], \qquad (10)$$

where $c(s_t, g)$ is a cost function associated with reaching g from s_t. Training f_θ ensures that the quasimetric structure is learned efficiently and provides meaningful goal-directed guidance.

V. Experiments

Our experiments will use three offline goal-conditioned tasks, aiming to answer the following questions:

1) **Quasimetric Guidance vs. Return-to-Go (RTG):** How does replacing RTG conditioning with quasimetric distances affect trajectory optimization and goal-reaching performance?
2) **Effectiveness of Different Quasimetric Models:** Which quasimetric model provides better generalization and planning capabilities?
3) **Impact of Loss Functions:** How do different loss functions (AWR vs DDPGBC) influence quasimetric learning and goal-reaching success?

A. Experimental Setup

We first describe our evaluation environments, shown in Fig.2. We evaluate QuaD in D4RL AntMaze [6], a suite of six goal-conditioned navigation tasks featuring an 8-DoF Ant robot navigating from a starting position to a goal location. These tasks require long-horizon planning and trajectory stitching, making them well-suited for evaluating quasimetric-based decision transformers. We evaluate QuaD against a comprehensive set of baselines spanning three primary offline reinforcement learning paradigms: behavior cloning, value-based methods, and sequence modeling approaches. For behavior cloning, we include Behavior Cloning (BC), a standard supervised learning method trained to replicate actions in the dataset without using any reward or goal information, and Goal-Conditioned Behavior Cloning (GCBC) [41], which conditions the policy on goal states to imitate goal-reaching behaviors without value estimation. Among value-based methods, we compare against TD3+BC [22], which augments the TD3 actor-critic framework with behavior cloning regularization to ensure stability in the offline setting; OneStepRL [42], which limits value updates to a single step to avoid extrapolation errors over long horizons; and Goal-Conditioned IQL (GC-IQL), an adaptation of Implicit Q-Learning [24] for goal-conditioned tasks that filters actions based on learned Q-values and avoids value overestimation. For sequence modeling, we include the Decision Transformer (DT) [5], which models trajectories autoregressively and conditions on return-to-go (RTG) to predict actions, and the Q-Learning Decision Transformer (QLDT), a variant of DT that incorporates Q-values into the transformer input to guide prediction toward high-value behaviors. All baselines are evaluated on the AntMaze benchmark under identical conditions using five random seeds, with 95% confidence intervals shown via shaded regions in figures or standard deviations reported in tables. Additional training and implementation details are provided in the Appendix.

B. Main Results on AntMaze Environments

Table I summarizes the success rates (%) and standard errors across multiple seeds, comparing our approach against various state-of-the-art offline RL methods, including TD3+BC [22], OneStep RL [42], BC (Behavior Cloning), and Decision Transformer (DT) [5]. The transformer-based methods (right side of the vertical line) are particularly relevant for comparing our approach, as they employ sequence modeling techniques.

a) Overall Performance Trends: Our methods, QuaD (IQE) and QuaD (MRN), significantly outperform Decision Transformer (DT) and QLDT in all environments, particularly in more complex mazes. While DT struggles to achieve meaningful success rates, our approach demonstrates robust performance even in difficult settings. Notably, on the easier **umaze** environments, QuaD (IQE) achieves a success rate of 91.0%, far surpassing DT (53.6%) and QLDT (67.2%). Similarly, in **umaze-diverse**, both IQE and MRN models reach 91.4%, outperforming all baselines.

b) Performance in Medium and Large Mazes: In more challenging medium and large mazes, our method significantly improves over prior approaches. Notably, in the medium-play setting, DT and QLDT both fail to achieve meaningful success rates, whereas our QuaD (IQE) and QuaD (MRN) models achieve 59.4% and 60.8% success rates, respectively, demonstrating the advantage of quasimetric-based distance guidance. Similarly, in medium-diverse, both of our models maintain a high success rate around 60%, while all prior transformer-based methods fail to solve the task.

c) Challenging Large Maze Tasks.: The **large-scale AntMaze tasks** remain among the most challenging bench-

Environment	TD3+BC	OneStepRL	BC	GCBC	GC-IQL	DT	QLDT	QuaD(IQE)	QuaD(MRN)
An-U-v2	78.6	64.3	54.6	67.3 ± 10.1	63.5 ± 14.6	53.6 ± 7.3	67.2 ± 2.3	**91.0 ± 3.16**	89.2 ± 3.82
An-UD-v2	71.4	60.7	45.6	71.9 ± 16.2	70.9 ± 11.2	42.2 ± 5.4	62.1 ± 1.6	**91.4 ± 3.58**	91.4 ± 3.23
An-MP-v2	10.6	0.3	0	20.2 ± 9.1	50.7 ± 18.8	0.0	0.0	59.4 ± 3.66	**60.8 ± 3.24**
An-MD-v2	3.0	0.0	0	23.1 ± 15.6	56.5 ± 14.4	0.0	0.0	**60.6 ± 2.87**	57.8 ± 3.2
An-LP-v2	0.2	0.0	0	14.4 ± 9.7	21.6 ± 15.2	0.0	0.0	**33.2 ± 3.80**	32.0 ± 1.79
An-LD-v2	0.0	0.0	0	20.7 ± 9.7	29.8 ± 12.4	0.0	0.0	**31.2 ± 2.07**	30.4 ± 3.36

TABLE I: **Offline RL benchmarks**: We use the AntMaze suite [6] of goal-conditioned RL tasks to compare our method to prior methods, measuring the success rate and standard error across multiple seeds. The methods on the right of the vertical line are transformer-based methods, the top scores among which are highlighted in **bold**. To save space, the name of the environments and datasets are abbreviated as follows: for the environments An=Ant; for the datasets U=umaze, UD=umaze-diverse, MP=medium-play, MD=medium-diverse, LP=large-play, LD-large-diverse. The proposed solution performs well.

marks in offline RL. While all prior transformer-based methods fail completely (DT and QLDT achieve 0% success), our models significantly outperform previous baselines, achieving 33.2% (IQE) and 32.0% (MRN) on large-play, and 31.2% (IQE) and 30.4% (MRN) on large-diverse. This demonstrates that our quasimetric distance-based approach enables effective long-horizon goal reaching, even in highly sparse-reward settings.

d) Comparison with Traditional Offline RL.: Traditional offline RL methods such as TD3+BC, OneStep RL, and BC fail to generalize effectively across AntMaze tasks. While TD3+BC achieves some success on umaze and umaze-diverse, its performance drops significantly in medium and large environments, where goal-conditioned trajectory stitching is required. Our method, on the other hand, maintains strong performance across all difficulty levels, highlighting its advantage in long-horizon tasks requiring strategic planning.

Overall, QuaD (IQE) and QuaD (MRN) consistently outperform DT, QLDT, and other prior methods across all AntMaze tasks. The results validate our hypothesis that replacing RTG with quasimetric guidance enables better goal-directed decision-making in sequence-based RL. Moreover, IQE slightly outperforms MRN in most settings, suggesting that interval-based quasimetric embeddings provide a stronger representation for long-horizon trajectory modeling. These findings establish QuaD as a powerful alternative to traditional RTG-based Decision Transformers, particularly in goal-conditioned RL.

C. Ablation Studies

We probe QuaD's design choices via ablations on (i) the quasimetric model and (ii) the learning loss, measuring success on six AntMaze tasks (Tables II, III).

1) Effectiveness of Different Quasimetric Methods: QuaD replaces RTG with a learned quasimetric. We compare:

- **Interval Quasimetric Embeddings (IQE):** interval-sorted state–goal embeddings aggregated by mean/max pooling, enforcing implicit order and robustness to trajectory perturbations.
- **Metric Residual Networks (MRN):** residual over Euclidean distance with an asymmetric ℓ_∞ term to capture directed transition dynamics.

Results. Across AntMaze:

1) **IQE vs. MRN (overall).** IQE generally leads, especially in structured mazes. In AntMaze-Umaze, IQE reaches **91.0% (DDPG+BC)** and **93.2% (AWR)** vs. MRN's **89.2% (DDPG+BC)** and **92.4% (AWR)**, indicating strong local trajectory stitching from IQE.
2) **Medium-scale planning.** Performance is close: in Medium-Play (DDPG+BC), **MRN 60.8%** vs. **IQE 59.4%**; in Medium-Diverse (AWR), **IQE 61.0%** vs. **MRN 57.8%**. MRN's residual helps with longer horizons, while IQE remains competitive.
3) **Long horizons (large mazes).** Both degrade under extreme sparsity/complexity. In Large-Play (DDPG+BC), **IQE 33.2%** vs. **MRN 32.0%**; in Large-Diverse, both converge to **31.2%**. Neither extrapolates strongly at extreme horizons.

Overall, IQE excels in small/medium settings; MRN gains stability on longer horizons. In large mazes, both plateau, motivating stronger quasimetric learning for extreme long-horizon goals.

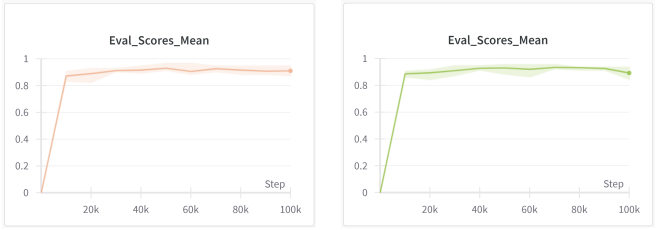

Fig. 3: QuaD learning curves on `antmaze-Umaze-v2` with IQE (left) and MRN (right).

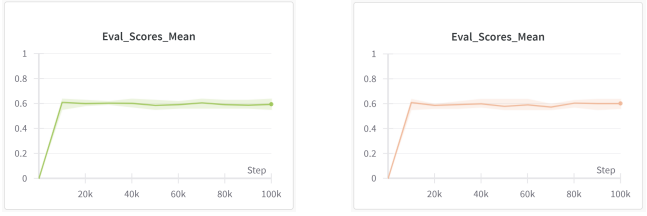

Fig. 4: QuaD learning curves on `antmaze-medium-play-v2` with IQE (left) and MRN (right).

2) Impact of Different Loss Functions: We study two losses shaping the quasimetric and policy:

- **AWR:** BC reweighted by an exponential advantage derived from the quasimetric; favors low-distance trajectories and stabilizes early learning.
- **DDPG+BC:** Hybrid offline Q-learning with BC regularization; curbs value overestimation and supports long-horizon stitching.

Results.

1) **Short-horizon, structured tasks.** In Umaze, IQE+AWR attains **93.2%** vs. **91.0%** for DDPG+BC; in Umaze-Diverse, AWR-based IQE **89.9%** vs. DDPG+BC **91.4%**. AWR promotes strong local decisions in well-defined settings.
2) **Medium/large tasks.** IQE with DDPG+BC slightly leads: Medium-Play **59.4%** (vs. **58.4%** AWR); Medium-Diverse **61.0%** (vs. **60.6%** AWR). The gap widens in Large-Play **33.2%** (vs. **31.2%** AWR); Large-Diverse **31.2%** for both. Q-learning improves long-horizon stitching.
3) **MRN trends.** MRN mirrors IQE but slightly lower overall: Umaze **92.4%** (AWR) and **89.2%** (DDPG+BC). In longer horizons MRN benefits notably from DDPG+BC (e.g., Medium-Play **60.8%**; Large-Play **32.0%**), narrowing the gap.

Takeaways. AWR yields stability and fast early progress (ideal for short, structured tasks), while DDPG+BC enhances long-horizon planning (medium/large mazes). IQE is strongest overall, and MRN gains more from DDPG+BC. An adaptive objective that blends AWR's stability with DDPG+BC's planning benefits is a promising direction.

Environment	IQE (AWR)	IQE (DDPG+BC)
An-U-v2	93.2 ± 3.21	91.0 ± 3.16
An-UD-v2	89.9 ± 3.23	91.4 ± 3.58
An-MP-v2	58.4 ± 3.66	59.4 ± 3.63
An-MD-v2	61.0 ± 2.07	60.6 ± 2.87
An-LP-v2	31.2 ± 2.28	33.2 ± 3.80
An-LD-v2	31.2 ± 2.17	31.2 ± 2.07

TABLE II: Success rate (%) with standard error for IQE using AWR loss and the DDPG+BC loss. Environments: An=Ant. Datasets: U=umaze, UD=umaze-diverse, MP=medium-play, MD=medium-diverse, LP=large-play, LD=large-diverse.

D. Summary of Ablation Findings

Our ablation studies provide key insights into the effectiveness of different quasimetric models, the impact of loss function selection, and the robustness of QuaD to quasimetric inaccuracies. The results from Tables II and III highlight the following key takeaways:

- IQE consistently outperforms MRN in structured environments but faces challenges in long-horizon tasks.

Environment	MRN (AWR)	MRN (DDPG+BC)
An-U-v2	92.4 ± 5.94	89.2 ± 3.82
An-UD-v2	89.3 ± 3.23	91.4 ± 3.23
An-MP-v2	57.2 ± 4.36	60.8 ± 3.24
An-MD-v2	58.6 ± 2.19	57.8 ± 3.2
An-LP-v2	28.4 ± 2.07	32.0 ± 1.79
An-LD-v2	31.2 ± 2.17	30.4 ± 3.36

TABLE III: Success rate (%) with standard error for MRN using AWR loss and the DDPG+BC loss. Environments: An=Ant. Datasets: U=umaze, UD=umaze-diverse, MP=medium-play, MD=medium-diverse, LP=large-play, LD=large-diverse.

- DDPG+BC significantly improves long-horizon planning and goal-reaching success, outperforming AWR in larger environments.
- DDPG+BC is the most effective loss function overall, achieving highest success rates across all AntMaze tasks.
- Quasimetric-based trajectory modeling provides a significant advantage over RTG-based Decision Transformers.

These findings emphasize the importance of quasimetric selection and loss function choice in effective trajectory modeling. Future improvements may focus on adaptive loss function strategies and hierarchical extensions that integrate quasimetric subgoal discovery for enhanced long-horizon planning.

VI. Conclusion

We introduced Quasimetric Decision Transformer (QuaD), a novel framework that replaces return-to-go (RTG) conditioning in Decision Transformers with learned quasimetric distances for goal-conditioned RL. By leveraging quasimetric learning, QuaD provides a structured, goal-aware signal that improves trajectory optimization, generalization to unseen goals, and long-horizon planning. Our experiments on AntMaze tasks demonstrate that QuaD significantly outperforms standard Decision Transformers across all settings, with IQE excelling in structured navigation tasks. We show that Advantage-Weighted Regression (AWR) is the most effective loss formulation, while DDPG+BC can further aid long-horizon trajectory stitching. Theoretical analysis confirms that quasimetric distances offer a superior success predictor compared to RTG, leading to more effective decision-making. This work establishes the first systematic study of metric learning in sequence-based RL, bridging the gap between Decision Transformers and distance-based goal representations. Future directions include hierarchical RL with quasimetric-based subgoal discovery, contrastive quasimetric learning, and real-world applications in robotics. By introducing quasimetric guidance in DTs, we open a new research avenue for scalable and structured goal-conditioned RL.

References

[1] A. E. Sallab, M. Abdou, E. Perot, and S. Yogamani, "Deep reinforcement learning framework for autonomous driving," Electronic

Imaging, vol. 29, no. 19, p. 70–76, Jan. 2017. [Online]. Available: http://dx.doi.org/10.2352/ISSN.2470-1173.2017.19.AVM-023

[2] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever et al., "Mastering the game of go with deep neural networks and tree search," Nature, vol. 529, no. 7587, pp. 484–489, Jan. 2016.

[3] S. Levine, A. Kumar, G. Tucker, and J. Fu, "Offline reinforcement learning: Tutorial, review, and perspectives on open problems," arXiv preprint arXiv:2005.01643, 2020.

[4] A. Kumar, A. Zhou, G. Tucker, and S. Levine, "Conservative q-learning for offline reinforcement learning," 2020. [Online]. Available: https://arxiv.org/abs/2006.04779

[5] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, "Decision transformer: Reinforcement learning via sequence modeling," Advances in neural information processing systems, vol. 34, pp. 15 084–15 097, 2021.

[6] J. Fu, A. Kumar, O. Nachum, G. Tucker, and S. Levine, "D4rl: Datasets for deep data-driven reinforcement learning," 2020.

[7] T. Wang and P. Isola, "Improved representation of asymmetrical distances with interval quasimetric embeddings," in NeurIPS 2022 Workshop on Symmetry and Geometry in Neural Representations, 2022. [Online]. Available: https://openreview.net/forum?id=KRiST_rzkGl

[8] B. Liu, Y. Feng, Q. Liu, and P. Stone, "Metric residual network for sample efficient goal-conditioned reinforcement learning," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, 2023, pp. 8799–8806.

[9] X. B. Peng, A. Kumar, G. Zhang, and S. Levine, "Advantage-weighted regression: Simple and scalable off-policy reinforcement learning," 2019. [Online]. Available: https://arxiv.org/abs/1910.00177

[10] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," in International Conference on Learning Representations, 2016. [Online]. Available: http://arxiv.org/abs/1509.02971

[11] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, O. Pieter Abbeel, and W. Zaremba, "Hindsight experience replay," Advances in neural information processing systems, vol. 30, 2017.

[12] B. Eysenbach, T. Zhang, S. Levine, and R. R. Salakhutdinov, "Contrastive learning as goal-conditioned reinforcement learning," Advances in Neural Information Processing Systems, vol. 35, pp. 35 603–35 620, 2022.

[13] R. Ghugare, M. Geist, G. Berseth, and B. Eysenbach, "Closing the gap between TD learning and supervised learning - a generalisation point of view." in The Twelfth International Conference on Learning Representations, 2024. [Online]. Available: https://openreview.net/forum?id=qg5JENs0N4

[14] S. Pateria, B. Subagdja, A.-h. Tan, and C. Quek, "Hierarchical reinforcement learning: A comprehensive survey," ACM Computing Surveys (CSUR), vol. 54, no. 5, pp. 1–35, 2021.

[15] E. Chane-Sane, C. Schmid, and I. Laptev, "Goal-conditioned reinforcement learning with imagined subgoals," in International conference on machine learning. PMLR, 2021, pp. 1430–1440.

[16] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," nips, 2017.

[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Annual Conference of the North American Chapter of the Association for Computational Linguistics, 2019.

[18] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in IEEE/CVF Conference on Computer Vision and Pattern Recognition. arXiv, 2022.

[19] M. Janner, Q. Li, and S. Levine, "Offline reinforcement learning as one big sequence modeling problem," Advances in neural information processing systems, vol. 34, pp. 1273–1286, 2021.

[20] K.-H. Lee, O. Nachum, M. S. Yang, L. Lee, D. Freeman, S. Guadarrama, I. Fischer, W. Xu, E. Jang, H. Michalewski et al., "Multi-game decision transformers," Advances in Neural Information Processing Systems, vol. 35, pp. 27 921–27 936, 2022.

[21] S. Reed, K. Zolna, E. Parisotto, S. G. Colmenarejo, A. Novikov, G. Barth-maron, M. Giménez, Y. Sulsky, J. Kay, J. T. Springenberg, T. Eccles, J. Bruce, A. Razavi, A. Edwards, N. Heess, Y. Chen, R. Hadsell, O. Vinyals, M. Bordbar, and N. de Freitas, "A generalist agent," Transactions on Machine Learning Research, 2022, featured Certification, Outstanding Certification. [Online]. Available: https://openreview.net/forum?id=1ikK0kHjvj

[22] S. Fujimoto and S. S. Gu, "A minimalist approach to offline reinforcement learning," Advances in neural information processing systems, vol. 34, pp. 20 132–20 145, 2021.

[23] S. Emmons, B. Eysenbach, I. Kostrikov, and S. Levine, "Rvs: What is essential for offline RL via supervised learning?" in International Conference on Learning Representations, 2022. [Online]. Available: https://openreview.net/forum?id=S874XAIpkR-

[24] I. Kostrikov, A. Nair, and S. Levine, "Offline reinforcement learning with implicit q-learning," in International Conference on Learning Representations, 2022. [Online]. Available: https://openreview.net/forum?id=68n2s9ZJWF8

[25] P. Dayan, "Improving generalization for temporal difference learning: The successor representation," Neural Computation, vol. 5, no. 4, pp. 613–624, 1993.

[26] A. Barreto, W. Dabney, R. Munos, J. J. Hunt, T. Schaul, H. P. van Hasselt, and D. Silver, "Successor features for transfer in reinforcement learning," Advances in neural information processing systems, vol. 30, 2017.

[27] C. J. Watkins and P. Dayan, "Q-learning," Machine learning, vol. 8, pp. 279–292, 1992.

[28] A. Touati and Y. Ollivier, "Learning One Representation to Optimize All Rewards," Oct. 2021.

[29] V. Micheli, K. Sinnathamby, and F. Fleuret, "Multi-task reinforcement learning with a planning quasi-metric," arXiv preprint arXiv:2002.03240, 2020.

[30] I. Durugkar, M. Tec, S. Niekum, and P. Stone, "Adversarial intrinsic motivation for reinforcement learning," in Advances in Neural Information Processing Systems, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., 2021. [Online]. Available: https://openreview.net/forum?id=GYr3qnFKgU

[31] P. Hansen-Estruch, A. Zhang, A. Nair, P. Yin, and S. Levine, "Bisimulation makes analogies in goal-conditioned reinforcement learning," in International Conference on Machine Learning. PMLR, 2022, pp. 8407–8426.

[32] N. Ferns, P. Panangaden, and D. Precup, "Bisimulation metrics for continuous markov decision processes," SIAM Journal on Computing, vol. 40, no. 6, pp. 1662–1714, 2011.

[33] T. Wang and P. Isola, "On the learning and learnability of quasimetrics," in International Conference on Learning Representations, 2022. [Online]. Available: https://openreview.net/forum?id=y0VvIg25yk

[34] D. Borsa, A. Barreto, J. Quan, D. Mankowitz, R. Munos, H. van Hasselt, D. Silver, and T. Schaul, "Universal successor features approximators," in International Conference on Learning Representations. arXiv, 2019. [Online]. Available: http://arxiv.org/abs/1812.07626

[35] A. Barreto, W. Dabney, R. Munos, J. J. Hunt, T. Schaul, H. van Hasselt, and D. Silver, "Successor Features for Transfer in Reinforcement Learning," Feb. 2022.

[36] L. Blier, C. Tallec, and Y. Ollivier, "Learning Successor States and Goal-Dependent Values: A Mathematical Viewpoint," Jan. 2021. [Online]. Available: http://arxiv.org/abs/2101.07123

[37] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever et al., "Language models are unsupervised multitask learners," OpenAI blog, vol. 1, no. 8, p. 9, 2019.

[38] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell et al., "Language models are few-shot learners," Advances in neural information processing systems, vol. 33, pp. 1877–1901, 2020.

[39] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann et al., "Palm: Scaling language modeling with pathways," Journal of Machine Learning Research, vol. 24, no. 240, pp. 1–113, 2023.

[40] T. Wang, A. Torralba, P. Isola, and A. Zhang, "Optimal goal-reaching reinforcement learning via quasimetric learning," in International Conference on Machine Learning. PMLR, 2023, pp. 36 411–36 430.

[41] D. Ghosh, A. Gupta, A. Reddy, J. Fu, C. Devin, B. Eysenbach, and S. Levine, "Learning to reach goals via iterated supervised learning," in International Conference on Learning Representations. arXiv, 2021. [Online]. Available: http://arxiv.org/abs/1912.06088

[42] D. Brandfonbrener, W. Whitney, R. Ranganath, and J. Bruna, "Offline rl without off-policy evaluation," Advances in neural information processing systems, vol. 34, pp. 4933–4946, 2021.