ON DISTRIBUTIONAL REINFORCEMENT LEARNING IN CHAOTIC DYNAMICAL SYSTEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Chaotic dynamical systems pose a fundamental challenge for Reinforcement Learning (RL): exponential sensitivity to initial conditions induces high-variance bootstrap targets and poorly conditioned gradient updates. Chaotic dynamics arise across scientific and engineering domains, from fluid flows and climate systems to multi-agent systems, where reliable learning is highly desirable. Standard RL methods optimise expected returns through scalar value functions, implicitly averaging over diverging trajectories and entangling trajectory level instability with the learning objective. We show that under mild statistical stability assumptions, the return distribution evolves more regularly than individual trajectories when measured under the 1-Wasserstein metric, yielding a smoother distributional Bellman objective. By aligning optimisation with this measure level structure, distributional RL provides better conditioned learning. We offer a principled explanation for the advantages of distributional methods in chaotic systems and the geometries of RL objectives under chaos.

1 INTRODUCTION

In many applications of Reinforcement Learning (RL), chaos as itself is oft ignored and is instead conflated with stochasticity, but this framing obscures some of its important structure. In fact, unlike stochastic noise, chaos is typically deterministic in nature, arising from well-defined underlying dynamical rules, yet unpredictable because of amplification of infinitesimal perturbations over time (Strogatz, 2024). The compounded errors lead to multiple qualitatively distinct long term outcomes and sets of possible system states that form intricate self-similar geometric structures rather than smooth landscapes (Mandelbrot, 2004). As a result, averaging returns, a standard assumption in RL, can be fundamentally misleading. The intersection of RL and chaotic dynamics is broad and increasingly prominent, spanning multi-agent systems (Sato et al., 2002; Rudd-Jones et al., 2026), the stabilisation of financial systems (Rigatos et al., 2019), turbulence reduction in aero- and hydrofoil flows (Montalà et al., 2025), climate policy design (Rudd-Jones et al., 2025b;a), and synchronisation of power grids (Eroglu et al., 2017; Halekotte et al., 2021). However, the characteristics of chaotic systems render many traditional RL assumptions invalid, including smoothness in value landscapes and stable gradient updates (Boccaletti et al., 2000; Wang et al., 2023; 2024).

We start with a vignette for the reader: consider two states separated by a singular floating point numerical precision unit in a chaotic system. Under identical open-loop policies, one trajectory reaches a goal with high reward while the other diverges into a failure basin. Standard RL collapses these outcomes into a single scalar value, producing a misleading target that corresponds to no realisable behaviour. In chaotic systems, infinitesimal errors in state estimation or action selection amplify exponentially over time, even tiny differences in state can lead to vastly different future trajectories which undermines reliable long-term prediction and control. This “butterfly effect" (Lorenz, 1972) creates a hostile optimisation landscape as we lack smoothness in state for value function representation and can receive a large variance of returns when trying to directly optimise policies. Traditional methods typically struggle because the bounds of value prediction error explode rapidly, rendering long-horizon planning inextricably noisy (Young & Pugeault, 2024).

We study the mismatch between chaotic dynamics and the optimisation objective used in standard RL, which typically maximises expected return via scalar value functions. In chaotic systems, due to the exponential divergence of trajectories, long-horizon returns become highly sensitive to initial

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.

conditions. Averaging over such trajectories produces irregular bootstrapped targets, leading to sharp curvature and severe gradient scaling issues in the optimisation landscape. Although individual trajectories diverge, the evolution of probability measures describing long-term system behaviour is often statistically regular (Ruelle, 1976; Bowen, 2008). Motivated by this observation, we study RL through the lens of return distributions and Wasserstein geometry: under mild statistical stability assumptions, the return distribution varies more regularly than trajectory level quantities, yielding a smoother optimisation objective and providing a principled explanation for the potential effectiveness of distributional RL in chaotic environments. Our contributions can be summarised as follows:

- We prove that for chaotic dynamics the distributional RL objective is smoother than the expectation based objective: under mild statistical stability assumptions the return distribution is Lipschitz continuous in the 1-Wasserstein metric even when trajectories diverge exponentially.
- We empirically analyse optimisation geometry in chaotic environments and show that distributional objectives produce smoother loss landscapes and lower variance one-step targets, in turn enforcing bounded gradient norms.
- We show that distributional Q-learning methods can improve converged episodic return for some canonical control in chaos experiments over a non-distributional approach.

2 Related Work

Model-Free Reinforcement Learning for Chaotic Systems. The optimisation landscape in chaotic environments is notoriously difficult: Wang et al. (2023) demonstrated that the value functions and policy optimisation landscape in chaotic systems is non-smooth, rendering standard gradient based methods unstable or inefficient. This behaviour stems from the non-smoothness in state caused by sensitivity to initial conditions and exponential divergence of trajectories. To address these irregularities, Wang et al. (2024) analysed the mollification effects of stochastic policies, showing that the combination of a fractal value function with the Gaussian policy kernel effectively smooths out the high frequencies. This theoretical insight explains why policy gradient methods can succeed in chaotic domains as the policy acts as an inherent low-pass filter. However, in some environments this smoothing impacts the global objective, causing non-distributional RL to fail (Wang et al., 2024).

Distributional Reinforcement Learning for Chaotic Systems. A canonical example of distributional RL to chaotic control is the autonomous navigation of stratospheric balloons studied by Bellemare et al. (2020). In this setting, the atmospheric wind field forms a chaotic dynamical system and horizontal control is achieved solely by adjusting altitude to exploit different wind currents. The authors employed QRDQN (Dabney et al., 2018), a distributional RL algorithm that exceeded non-distributional approaches, attributed in part to the multimodal return distribution induced by chaotic winds. Our work can be seen as providing theoretical justification for the empirical success of distributional RL observed by Bellemare et al. (2020) for control under chaos.

Smoothness in Distributional Reinforcement Learning. While standard RL optimises the expected return, distributional RL models the full return distribution (Bellemare et al., 2017). Beyond improved representational capacity, prior work shows that distributional methods can yield more favourable optimisation properties than expectation based approaches (Sun et al., 2022). In particular, the distributional Bellman operator is Lipschitz continuous under metrics such as the Wasserstein distance (Bellemare et al., 2017) and the Cramér distance (Rowland et al., 2018), even when the expected Bellman operator may fail to contract. These results suggest that distributional RL implicitly smooths the optimisation landscape by aggregating information across the return distribution. Such smoothing is reminiscent of the mollification observed in policy gradient methods and may help explain the empirical robustness of distributional RL in high variance settings such as chaotic environments. However, to the best of our knowledge, the interaction between distributional RL and chaotic dynamics has not previously been analysed.

3 Chaotic Dynamics and their Implications for Reinforcement Learning

In the section we provide an overview of our study, introducing chaotic dynamics and their impact on Reinforcement Learning (RL). We discuss how chaos is typically approached from a probabilistic

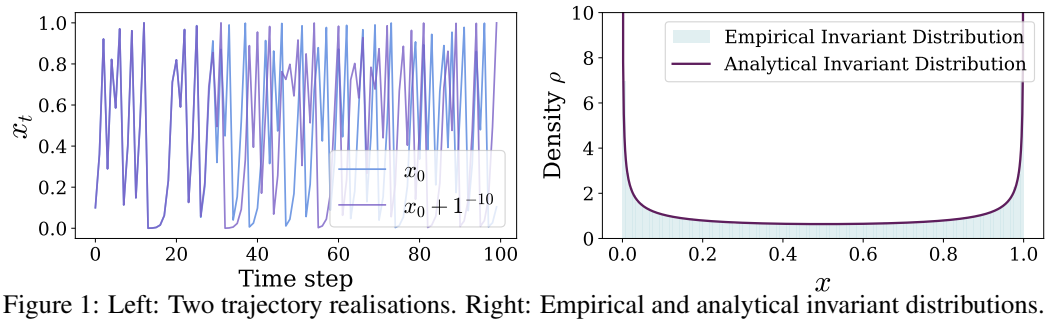

Figure 1: Left: Two trajectory realisations. Right: Empirical and analytical invariant distributions.

perspective, drawing parallels with learning the full distribution of returns in distributional RL. We posit that distributional RL mitigates several failure modes that arise in chaotic environments, where expectation-based value learning often struggles. However, we observe that distributional RL may not be sufficient on its own to tackle the full range of challenges considered in all scenarios.

3.1 What Does Chaos Mean Mathematically?

Non-linear dynamics frequently exhibit chaotic behaviour under certain parameters; chaos is a ubiquitous phenomenon across many natural processes (Strogatz, 2024). Formally, a (continuous-time) dynamical system is defined on a phase space M, a smooth manifold where every possible state of the system corresponds to a unique point $s \in M$. The system evolves according to a flow $\phi_t : M \to M$ such that $s(t) = \phi_t(s_0)$ describes the trajectory from initial condition s_0 given time t. Chaos is characterised by an extreme sensitivity to initial conditions (Strogatz, 2024). For two infinitesimally close initial states s_0 and $s_0 + \delta s_0$ the distance between their trajectories at time t separates exponentially as $\|s'(t) - s(t)\| \approx e^{\lambda t}\|\delta s_0\|$, where $s(t)$ is the trajectory starting at s_0 and $s'(t)$ the trajectory starting at $s_0 + \delta s_0$, quantified by the Lyapunov exponent λ (Wolf et al., 1985; Shimada & Nagashima, 1979). With maximal Lyapunov exponent $\lambda_{\max} = \lim_{t\to\infty} \lim_{\|\delta s_0\|\to 0} \frac{1}{t} \ln \frac{\|\delta s(t)\|}{\|\delta s_0\|}$. Systems with $\lambda_{\max} > 0$ exhibit exponential error amplification, are typically regarded chaotic, imply that long term prediction becomes exponentially sensitive to initial conditions despite deterministic governing equations. The reciprocal of the maximal Lyapunov exponent, known as Lyapunov time $T_\lambda \approx 1/\lambda_{\max}$, represents the horizon beyond which state predictions become effectively meaningless due to the exponential amplification of infinitesimal errors.

3.2 Why Distributions are More Stable than Trajectories Under Chaotic Dynamics

Dissipative dynamical systems typically evolve toward an attractor, an invariant subset of the phase space (Strogatz, 2024). While simple systems admit fixed points or limit cycles, chaotic systems often possess strange attractors, which exhibit fractal geometry and thus a non-integer Hausdorff dimension (Ruelle, 1976). Interestingly, although trajectories diverge exponentially, the strange attractor is a bounded set in phase space that confines trajectories, within which they exhibit chaotic behaviour.

Strange attractors admit invariant probability measures, often Sinai-Ruelle-Bowen (SRB) measures, which characterise the system's long term statistical behaviour (Sinai, 1972; Ruelle, 1976; Bowen, 2008). Modelling this invariant distribution is substantially more robust than predicting individual trajectories as beyond Lyapunov time arbitrarily small errors are exponentially amplified, making long-horizon accuracy impossible (Jiang et al., 2023; Schiff et al., 2024). In contrast, learning the geometry and statistics of the attractor captures physically valid long term behaviour.

The Logistic Map, a canonical example of discrete time chaos illustrates this distinction, defined as $s_{t+1} = ms_t(1 - s_t)$ where m adjusts the mapping's chaotic nature. Infinitesimally close initial conditions quickly decorrelate (Figure 1 Left), yet the empirical distribution of visited states converges to a stable invariant density (Figure 1 Right). Thus, although trajectories diverge exponentially and are individually unpredictable, the map admits a stable invariant measure, so that long-run time averages and distributional properties converge.

3.3 Challenges of Chaotic Transitions in Reinforcement Learning

Recent findings by Wang et al. (2023) demonstrate that in chaotic environments, value functions and consequently the policy gradient landscape exhibit fractal structure. In chaotic systems, because the return is defined through long-horizon rollouts, the state-action value function $Q(s, a)$ becomes highly sensitive to small changes in initial conditions and may be non-smooth with respect to state.

Such non-smoothness directly impacts gradient based optimisation. Temporal difference (TD) methods rely on bootstrapped targets of the form $r + \gamma \max_{a'} Q(s', a')$, irregularities in Q propagate through the Bellman recursion. Small state differences produce large target variations, increasing gradient variance and inducing sharp curvature in the loss landscape. In these regimes, local gradient information provides limited guidance for global progress, optimisation is highly sensitive to step size and initial conditions, and convergence may stall or oscillate near the optimum. These issues arise from properties of the underlying dynamical system.

Distributional RL (Bellemare et al., 2017) reflects this perspective by modelling the full return distribution rather than its expectation. The distributional Bellman operator is a contraction in the 1-Wasserstein metric, providing a well defined fixed point in a probability space (Bellemare et al., 2017). By optimising distances between return distributions, rather than scalar value errors, distributional methods shift the objective to a metric space with smoother geometry, potentially yielding a better conditioned optimisation landscape. In this sense, treating chaotic transitions probabilistically principally aligns the learning objective with the intrinsic instability of the dynamics.

4 How is Distributional Reinforcement Learning Beneficial for Chaotic Dynamics?

We model the environment as a time-homogeneous Markov Decision Process (MDP) defined by the tuple $\langle \mathcal{S}, \mathcal{A}, P, R, \mu_0, \gamma \rangle$ where $\mathcal{S}$ and $\mathcal{A}$ denote the continuous state and finite action spaces, $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ the transition dynamics, $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ the reward function, μ_0 the initial state distribution, and $\gamma \in (0, 1)$ the discount factor. At each timestep t an agent selects action $a_t \sim \pi(\cdot \mid s_t)$, the MDP transitions to the next state $s_{t+1} \in \mathcal{S}$ with probability $P(s_{t+1}|s_t, a_t)$ and in doing so receives reward $R(s_t, a_t)$. Actions are selected by the policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$. We maximise the discounted return $Z^\pi = \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)$. The state value and state-action value functions:

$$V^\pi(s) = \mathbb{E}_{a\sim\pi,\, s'\sim P}\left[R(s, a) + \gamma V^\pi(s')\right], \qquad V^\pi(s) = \mathbb{E}_\pi\left[Z^\pi \mid s_0 = s\right], \tag{1}$$

$$Q^\pi(s, a) = \mathbb{E}_{s'\sim P,\, a'\sim\pi}\left[R(s, a) + \gamma Q^\pi(s', a')\right], \quad Q^\pi(s, a) = \mathbb{E}_\pi\left[Z^\pi \mid s_0 = s, a_0 = a\right]. \tag{2}$$

Non-distributional RL estimates expectations of returns; in contrast, Distributional RL models the full return distribution. Let $Z(s, a)$ denote the random return obtained from state action pair (s, a). The distributional Bellman equation is $Z(s, a) \overset{D}{=} R(s, a) + \gamma Z(S', A')$, where $S' \sim P(\cdot \mid s, a)$, $A' \sim \pi(\cdot \mid S')$, and $\overset{D}{=}$ denotes distribution equality. Distributional RL minimises distances between return distributions under metrics such as Wasserstein, yielding contraction properties of the distributional Bellman operator, ensuring stable convergence (Bellemare et al., 2017).

4.1 A Distributional Objective with Wasserstein can be Smoother under Chaos

We argue that in chaotic dynamical systems, expectation based objectives lead to challenging gradient updates and highly irregular value functions (Wang et al., 2023). In contrast, a distributional objective using the Wasserstein metric yields smoother learning targets. To clarify, *we do not claim that the true return distribution is smooth in state*. Instability in expectation based methods occurs at the level of individual trajectories, even though statistical properties of chaotic systems can remain stable under small perturbations. This distinction motivates distributional RL, comparing return distributions under Wasserstein induces metric regularity absent from expectation based approaches. Consequently, although the mapping $s \mapsto Z^\pi(s, a)$ may be highly irregular at individual points, it can still be Lipschitz continuous as a mapping into the space of probability measures endowed with W_1, provided the transition dynamics are statistically stable. We formalise this result under the assumptions below:

Assumption 4.1 (One-Step Lipschitz Dynamics)**.** Let π be a Lipschitz continuous policy. We assume the closed-loop one-step dynamics $s_{t+1} = f(s_t, \pi(s_t))$ are K_f-Lipschitz continuous:

$$\|f(s_1, \pi(s_1)) - f(s_2, \pi(s_2))\| \le K_f \|s_1 - s_2\|, \tag{3}$$

with $K_f > 1$ in chaotic regimes. This assumption quantifies how perturbations in the state propagate through the dynamics. In chaotic regimes ($K_f > 1$), small differences in initial states lead to exponential trajectory divergence. Note that traditional Q-learning relies upon a greedy policy: a discontinuous step function and thus not Lipschitz continuous. By assuming Lipschitz continuity (e.g., Softmax policy or Gaussian policy as in PPO) we establish a lower bound on the instability.

Assumption 4.2 (Reward Regularity)**.** For all $a \in \mathcal{A}$ the reward function is K_R-Lipschitz continuous in state and rewards are bounded:

$$|R(s_1, a) - R(s_2, a)| \leq K_R \|s_1 - s_2\|. \tag{4}$$

This assumption ensures that small perturbations in the state lead to controlled changes in reward. Such smoothness is typical in problems where rewards correspond to continuous cost functions.

Assumption 4.3 (Wasserstein Statistical Stability of One-Step Transitions)**.** For each $a \in \mathcal{A}$ the transition distribution $P(\cdot|s, a)$ is K_P-Lipschitz continuous with respect to the Wasserstein-1 metric W_1. That is for any a:

$$W_1\left(P(\cdot|s_1, a), P(\cdot|s_2, a)\right) \leq K_P \|s_1 - s_2\|. \tag{5}$$

This assumption captures statistical regularity of the dynamics: although nearby trajectories may diverge, the resulting state distributions remain close in Wasserstein distance. While purely deterministic and perfectly observed chaotic systems may violate Assumption 4.3, most pragmatic RL problems introduce effective stochasticity through process noise, exploration noise, partial observability, state discretisation, sensor uncertainty, etc. Under these conditions the transition kernel becomes non-degenerate and varies smoothly in Wasserstein distance. Fully deterministic chaotic systems represent a special limiting case, whereas statistically stability is the norm in practical control (Viana, 1997).

Assumption 4.4 (Discount Condition for Distributional Stability)**.** We assume that:

$$\gamma K_P < 1 \tag{6}$$

This condition ensures that the Bellman operator is contractive under distributional perturbations. Intuitively, discounting prevents transition sensitivity from amplifying errors over long horizons.

Theorem 4.5 (One-Step Sensitivity of Scalar Returns)**.** *Under Assumptions 4.1 and 4.2, for any closed-loop policy π, the truncated return map $s \mapsto G_T(s)$ is Lipschitz (*$\mathrm{Lip}(G_T)$*) with constant:*

$$\mathrm{Lip}(G_T) \;\leq\; K_R \sum_{t=0}^{T-1} (\gamma K_f)^t \;=\; K_R \cdot \frac{(\gamma K_f)^T - 1}{\gamma K_f - 1} \quad (\gamma K_f \neq 1). \tag{7}$$

In particular, if $\gamma K_f > 1$, then $\mathrm{Lip}(G_T)$ grows exponentially in T. Proof in Appendix B.

In the theorem above we show that one-step samples under chaotic dynamics can explode.

Theorem 4.6 (W_1-smoothness of the return distribution under statistical stability)**.** *Under Assumptions 4.2 - 4.4, for any fixed initial action a the following holds:*

$$W_1(Z^\pi(s_1, a), Z^\pi(s_2, a)) \leq \frac{K_R}{1 - \gamma K_P} \|s_1 - s_2\|. \tag{8}$$

Proof in Appendix C.

Theorem 4.6 establishes that, under Assumption 4.3, the true return distribution $Z^\pi(s, a)$ varies Lipschitz continuously with respect to state when measured under 1-Wasserstein. We emphasise that the expectation of a Lipschitz distribution is itself Lipschitz. Thus Theorem 4.6 does not imply that expectation based value functions are non-smooth. Rather, when viewed as mappings into the space of probability measures equipped with W_1, return distributions vary in a stable manner across nearby states. The real advantage of distributional RL is that the distributional optimisation objective has better conditioning than scalar TD errors. Chaotic systems may exhibit exponential divergence at the trajectory level, but their induced evolution of probability measures can remain stable. Distributional RL leverages this measure level regularity rather than suppressing trajectory level chaos. In Appendix D we discuss in greater detail how distributional RL still copes when 4.3 fails.

Remark 4.7. It is crucial to note that Theorem 4.5 and 4.6 evaluate the exact same closed-loop system, yet yield vastly different stability guarantees. Theorem 4.5 relies on the trajectory expansion rate K_f. In chaotic systems, local exponential divergence guarantees that $K_f > 1$, forcing the scalar return to explode. Conversely, Theorem 4.6 relies on K_P, the Lipschitz constant of the transition kernel in the 1-Wasserstein metric. A defining feature of chaotic strange attractors is that while individual trajectories diverge exponentially ($K_f > 1$), their invariant probability measures are statistically stable. Therefore, it is consistent for a chaotic system to simultaneously exhibit $K_f > 1/\gamma$ (breaking scalar value smoothness) and $K_P < 1/\gamma$ (preserving distributional smoothness).

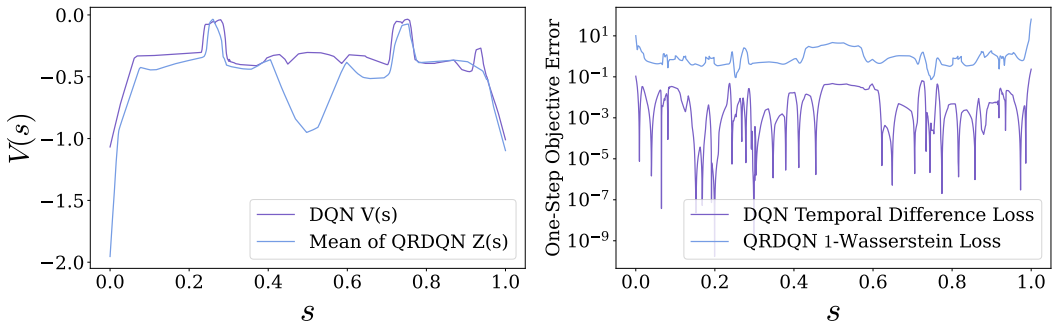

Figure 2: Left: $Q(s, a)$ for Action 1 for the Logistic Map, we take the mean of the QRDQN value function for a 1D representation. Right: Comparison of one-step errors between DQN and QRDQN. Figure data collected from agents after $3e8$ training steps.

Remark 4.8. If the environment is fully observed and deterministic, then Assumption 4.3 reduces to the dynamics Lipschitz constant, so $K_P = K_f$. In this regime $\gamma K_P < 1$ reduces to $\gamma K_f < 1$ which fails in chaotic systems ($K_f > 1$) and W_1-smoothness does not improve over one-step sensitivity. The regime where distributional smoothing becomes meaningful is when the effective transition kernel is non-degenerate which is very common in pragmatic RL. In such cases the kernel can be statistically stable in W_1 with $K_P \ll K_f$ even when trajectories exhibit strong local expansion.

5 Empirical Analysis

Above we have given a theoretical proof for the smoothness of the distributional RL objective under chaos. In the following section we experientially support these claims. We make a distinction of two types of chaotic environments: (i) settings in which the agent directly influences and stabilises the underlying chaotic system, and (ii) settings in which the chaotic dynamics are intrinsic to the environment and cannot be eliminated, but nevertheless influence the state evolution and observed returns. We experiment on both types to understand the benefits distributional methods bring.

5.1 Experimental Settings

Environments. For the former we look to control the Logistic Map, a canonical discrete time 1D chaotic map. Following Gadaleta & Dangelmayr (1999) the Logistic Map control task is defined as $x_{t+1} = (m + a_t)x_t(1 - x_t)$, where $m = 3.8$ is a parameter that adjusts the chaotic nature of the equation, $a \in [-0.1, 0.1]$ that perturbs the chaotic parameter, with arbitrary distance metric reward $r = \|x_t - x_{fp}\|^2$ to incentive an agent to stabilise the system to an unstable 1-periodic fixed point $x_{fp} = \frac{m-1}{m}$. We also use the Ikeda Map (Ikeda, 1979), a discrete time 2D chaotic map defined as:

$$x_{t+1} = 1 + (u + a_t)(x_t \cos \xi_t - y_t \sin \xi_t) \tag{9}$$

$$y_{t+1} = (u + a_t)(x_t \sin \xi_t + y_t \cos \xi_t), \tag{10}$$

where $\xi_t = k - \frac{p}{1+x_t^2+y_t^2}$, u an equation parameter (not control term), $a \in [-0.1, 0.1]$, the 2D state vector $s_t = [x_t, y_t]^\intercal \in \mathbb{R}^2$, with arbitrary distance metric reward $r = \|s_t - s_{fp}\|_2^2$ that incentives an agent to stabilise the system to a 1-periodic fixed point $s_{fp} = [0.533, 0.247]$. Chaotic behaviour is typical when $u = 0.9$, $k = 0.4$, and $p = 6.0$.

For the latter we use two continuous time chaotic flow based environments in which a "swimmer" must navigate towards a desired area of the state. In a similar manner to Bellemare et al. (2020) the swimmer is under-actuated, unable to move directly toward the goal, and thus must *understand and work with the chaos*. For a 2D scenario we use the canonical Double Gyre flow, that simulates simplified ocean currents, and a swimmer therein (Sulalitha Priyankara et al., 2017), defined as:

$$\dot{x} = -\pi A \sin(\pi f(x, t)) \cos(\pi y) \tag{11}$$

$$\dot{y} = \pi A \cos(\pi f(x, t)) \sin(\pi y) \frac{\partial f}{\partial x}, \tag{12}$$

where $f(x, t) = a(t)x^2 + b(t)x$, $a(t) = \zeta \sin(\omega t)$, $b(t) = 1 - 2\zeta \sin(\omega t)$, $x \in [0, 2], y \in [0, 1]$, A the flow magnitude, ω adjusts oscillation frequency, and ζ adjusts oscillation amplitude in x. Setting

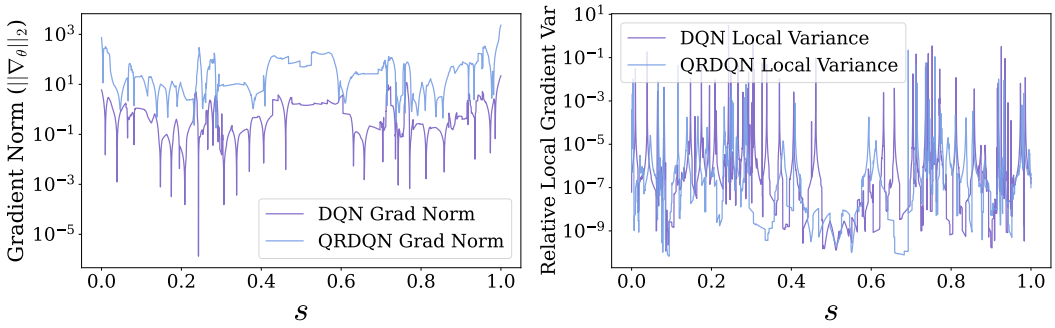

Figure 3: Left: Gradient norms ($||\nabla_\theta||_2$) of DQN and QRDQN for the Logistic Map. Right: Local gradient variance of DQN and QRDQN. Figure data collected from agents after $3e8$ training steps.

$\zeta = 0$ creates a time homogeneous system that is time inhomogeneous otherwise. We also use the 3D Arnold–Beltrami–Childress (ABC) flow (Qin & Liao, 2023), defined as:

$$\dot{x} = A \sin z + C \cos y \tag{13}$$
$$\dot{y} = B \sin x + A \cos z \tag{14}$$
$$\dot{z} = C \sin y + B \cos x. \tag{15}$$

For both environments we set a reward of $r(s) = -0.01 + 10 \cdot \exp\left(-d(s)^2/2\epsilon^2\right)$, where $d(s) = \|s - s_{\text{goal}}\|$, to give a smooth Gaussian peak within ϵ of the goal state to ensure Assumption 4.2.

Algorithms. For our distributional RL algorithm we chose QRDQN (Dabney et al., 2018) that uses a quantile regression loss which is equivalent to minimising the 1-Wasserstein distance (Dabney et al., 2018). The non-distributional method of choice was DQN (Mnih et al., 2013).

Description of the Experiments. We primarily focus on plotting the one-step error between distributional and non-distributional methods to compare smoothness in state. To explicitly evaluate optimisation conditioning, we calculate the L_2 gradient norm ($||\nabla_\theta \mathcal{L}||_2$) and the relative local gradient variance across the state space for the 1D Logistic Map.

5.2 Results

In Figure 2 we compare the landscapes of the value function and one-step error between DQN and QRDQN. On the left it is clear that both learn a similar representation of the environment, where we have flattened the QRDQN distribution by taking the mean over the quantiles. On the right houses some important results, the DQN temporal difference loss exhibits high variance and frequent sharp spikes, whereas the QRDQN 1-Wasserstein loss remains comparatively smoother and more stable across states: suggesting that the distributional objective provides an improved optimisation signal. In Appendix E we show the full QRDQN return distribution for the Logistic Map. The downstream impacts of this better behaved optimisation signal are seen in Figure 3. On the right, the local gradient variance remains complex for both DQN and QRDQN; the value functions are inherently non-smooth and thus the neural network's gradients will always show localised variance spikes when trying to fit these high-frequencies (Wang et al., 2023). However, on the left the gradient norm for QRDQN is bounded and maintains a consistent, non-vanishing range. QRDQN gradient norms are naturally larger since the neural network outputs N distinct quantiles rather than a singular scalar, but what is important is that the range of the norms is bounded. DQN gradient norms fluctuate over a larger range, even reaching a minimal value of $\sim 10^{-7}$.

In further support, we display the objective for the Double Gyre flow in Figure 4. Interestingly in Figure 4 one can see the goal state in the top right corner for both algorithms, but for QRDQN the one-step loss is much smoother over the state space with a much reduced error range. Since most trajectories accumulate similar step penalties before either reaching the goal or failing, the induced return distributions differ only minimally across nearby states. The Wasserstein objective correctly captures this measure level similarity, yielding a smooth and well conditioned landscape. In contrast, the scalar TD loss of DQN exhibits substantial variability due to bootstrap amplification of local estimation noise. Similar plots for the Ikeda Map can be found at Figure 6 in Appendix E.

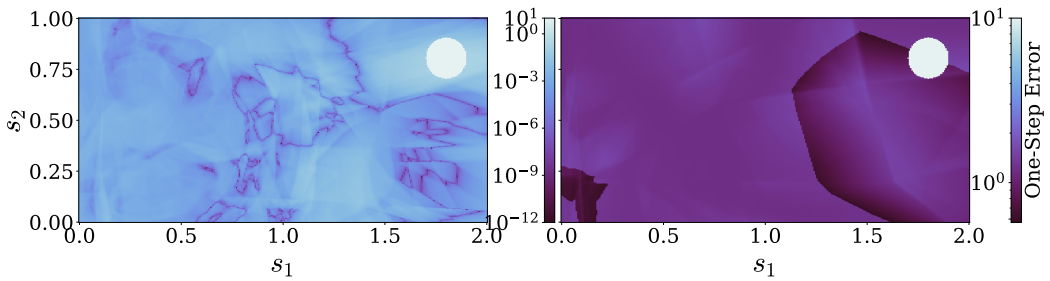

Figure 4: Double Gyre Flow objective landscapes. Left: DQN, Right: QRDQN. Maximal error has been clipped from 10^3 to 10^1 for clarity. Figure data collected from agents after $3e8$ training steps.

5.3 Discussion

While both DQN and QRDQN approximate value functions over chaotic dynamics, the optimisation geometry differs markedly. The fractal structure of the value function is unavoidable, as it reflects the intrinsic properties of the underlying chaotic dynamical system, and thus gradient variance is naturally also non-smooth. In Theorem 4.5 we proved that for scalar returns in chaotic regimes the Lipschitz constant grows exponentially; the downstream consequence of which can be vanishing/exploding gradients. Further, Theorem 4.6 proved that under statistical stability, the return distribution is W_1-Lipschitz continuous with a strict, finite bound. Figure 3 corroborates these insights: DQN suffers from gradient collapse, QRDQN maintains a bounded non-vanishing gradient norm. We emphasise that a smoother objective does not guarantee improved asymptotic performance; instead it improves conditioning of gradient updates and mitigates the impact of extreme bootstrap targets.

6 Benchmark Evaluation

We have shown theoretically and empirically the smoother objective landscape that distributional RL provides in chaotic systems. To finally exemplify our hypothesis, we compare distributional and non-distributional methods on the four aforementioned canonical chaotic control environments.

6.1 Experimental Settings

We track episodic return over training time, with each environment having a maximum number of steps. In the Logistic and Ikeda Map environments, episodes may also terminate once the system stabilises within a specified error of the fixed point. For the Double Gyre and ABC Flow environments, episodes terminate when the agent reaches the goal. Algorithms are evaluated using six random seeds, and we plot mean episodic return across runs with $10\% - 90\%$ quantile range. Further, we include PPO (Schulman et al., 2017) as a policy gradient comparison. We understand these are fairly simple environments in state space size, however show that RL methods still struggle here. All experiments are run on an Nvidia RTX 4090 with each run taking ≈ 6 hours, requiring ≈ 20GB of VRAM.

6.2 Results

Figure 5 compares DQN, QRDQN, and PPO across the four environments. In the first two environments (Logistic Map and Ikeda Map), the dynamics become less chaotic over time as they are effectively stabilised through the learning process. As a result, return distributions narrow and the advantage of modelling the full return distribution, as done in distributional RL, becomes less pronounced. We can see this in our results as QRDQN shows only marginal or inconsistent improvements over DQN. In contrast, in the latter environments (Double Gyre Flow and ABC Flow), strong chaotic behaviour persists throughout training. For the Double Gyre flow, QRDQN has an unstable early training period but eventually settles upon an optimal policy with a low variance between seeds. For the ABC flow, QRDQN eventually learns an improved policy but is unable to reach the same maximal return as PPO. In these settings, the ability of QRDQN to capture and learn from the full distribution of possible outcomes provides a clearer advantage, leading to more stable and superior performance. In all experiments, PPO is much more sample efficient reaching convergence in $1/5$ of the time it takes the other two. Further, the results from PPO set an upper bound for the maximal episodic return in all experiments but the Logistic Map.

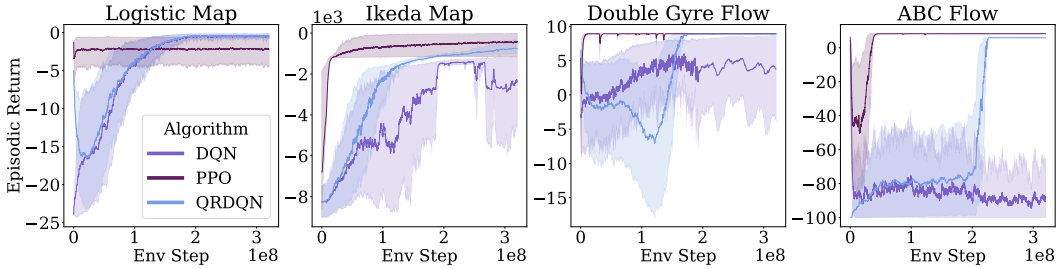

Figure 5: Each algorithm is run over six random seeds, we plot the mean episodic return across runs with shaded regions indicating the $10\% - 90\%$ quantile range at each environment step.

6.3 Discussion

Although in the Logistic and Ikeda Map the agent progressively suppresses chaotic behaviour by stabilising the system to a fixed point, even in the Double Gyre and ABC flow learning progressively regularises the experienced dynamics. As policies become more consistent, the state visitation distribution concentrates on narrower regions of the attractor. In this sense, learning induces a form of ordering even when the underlying flow remains chaotic. However, return stochasticity persists due to sensitivity to initial conditions, noise, and perturbations. Distributional methods remain useful because they capture this residual multi-modality instead of collapsing outcomes into an expectation.

From Figure 5 it is clear that PPO performs competitively across tasks. This aligns with the analysis of Wang et al. (2024) where Gaussian policies act as implicit mollifiers of fractal value landscapes. For Q-learning approaches the inherent challenge is that the policy is typically a step function, whereas in PPO we have a Lipschitz continuous policy. Policy gradients smooth via action stochasticity, while distributional RL smooths via probability geometry (e.g. Wasserstein distance). These are distinct but related mechanisms, reinforcing the importance of optimisation smoothing in chaotic systems.

In persistent chaos environments (Double Gyre and ABC Flow), QRDQN eventually reaches a similar episodic return as PPO but requires many more samples. This suggests distributional RL primarily improves optimisation stability (e.g. stabilised targets, reduced gradient variance) but has no effect on data efficiency. We therefore frame its benefit as improved conditioning and robustness in highly sensitive regimes for Q-learning based methods, but at least in the experiments we have used, the mollification effects of PPO are more impactful for sample efficiency and maximal return.

7 Conclusion

Summary of the Contributions. In this work we investigated how chaotic dynamics impact Q-learning based RL objectives. While chaos causes exponential trajectory sensitivity and non-smooth value landscapes, we show that return distributions can remain Lipschitz in Wasserstein distance under mild statistical stability assumptions. This measure level regularity helps explain why distributional RL produces better conditioned optimisation in chaotic environments. Empirically, distributional methods reduce loss variability and stabilise training. Gains are modest when chaos is suppressed, but clearer when strong mixing persists. We do not claim universal superiority rather, distributional RL offers robustness in settings with unstable bootstrap targets for Q-learning based methodologies.

Limitations. Our theory assumes statistical stability in Wasserstein distance, which may fail near bifurcations or in perfectly deterministic systems. Although distributional bounds may break down, modelling multi-modal returns can still outperform averaging. Our experiments focus on discrete actions; extending the analysis to continuous control remains important. Finally, our results emphasise optimisation stability rather than asymptotic optimality or sample efficiency improvements.

Future Work. Future directions include extending theory to continuous actions or studying representation learning with attractor geometry. For example, using distributional methods with policy gradient approaches would indicate whether the benefits of learning a distribution carry on over, perhaps the combination of mollification and learning a return distribution would be an optimal methodology for chaotic systems. Distributional RL and policy gradient methods apply smoothing, but still learn a fractal value function, can one avoid this tough representation task? Finally, recent work suggests alternative measures that better characterise chaotic dynamics (Finzi et al., 2026).

AI USE STATEMENT

In this work, we used generative AI tools for providing feedback on research methodology or experiments. We have not used generative AI tools for: help developing theoretical models or conceptual frameworks; formulating mathematical claims; providing critical ingredients for proving mathematical claims; assisting in the writing of proofs; proposing or refining hypotheses; designing research methodology or experiments; implementing methods; supporting qualitative and thematic data analysis; or interpreting results, and the tasks of: generating synthetic data sets; assisting with translation; and cleaning and reformatting a dataset are not applicable to this work. Additionally, we used generative AI tools for: summarising or analysing existing literature, sourcing/searching for information, and identifying any relevant literature. We have reviewed all AI-assisted work. We used LLM based literature review software to find all relevant literature for our ideas and hypotheses to ensure we covered all the bases. We used an LLM as another point of view when reviewing our research methodology and experiments, it provided valuable feedback in the idea iteration process. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

- **Theoretical Claims:** All underlying dynamical assumptions (Assumptions 4.1–4.4) are stated explicitly in Section 4. Complete mathematical proofs for the exponential sensitivity of scalar returns (Theorem 4.5) and the W_1-smoothness of return distributions (Theorem 4.6) are provided in Appendix B and Appendix C, respectively. Further analysis regarding the breakdown of statistical stability is detailed in Appendix D.
- **Environment Formulations:** Mathematical formulations, governing ODEs/mappings, boundary conditions, and continuous reward functions for all four benchmarks (Logistic Map, Ikeda Map, Double Gyre Flow, and ABC Flow) are fully defined in Section 5.1.
- **Implementation & Hyperparameters:** Network architectures, layer dimensions, activations, and comprehensive training hyperparameter configurations for DQN, QRDQN, and PPO are provided in Appendix G. Hardware specifications (Nvidia RTX 4090, 20GB VRAM) and compute times are described in Section 6.1.
- **Statistical Evaluation:** Benchmark comparisons in Section 6.2 are evaluated over six distinct random seeds, with reported mean performance and 10%–90% quantile intervals.
- **Code Availability:** As noted in Appendix A, all experimental code and chaotic benchmark environments (implemented in JAX) will be made publicly available as an open-source GitHub package upon acceptance.

REFERENCES

Marc G. Bellemare, Will Dabney, and Rémi Munos. A Distributional Perspective on Reinforcement Learning. In *Proceedings of the 34th International Conference on Machine Learning (ICML'17)*, 2017.

Marc G Bellemare, Salvatore Candido, Pablo Samuel Castro, Jun Gong, Marlos C Machado, Subhodeep Moitra, Sameera S Ponda, and Ziyu Wang. Autonomous navigation of stratospheric balloons using reinforcement learning. *Nature*, 588(7836):77–82, 2020.

Stefanos Boccaletti, Celso Grebogi, Y-C Lai, Hector Mancini, and Diego Maza. The control of chaos: theory and applications. *Physics Reports*, 329(3):103–197, 2000.

Rufus Bowen. Ergodic theory of axiom a diffeomorphisms. In *Equilibrium states and the ergodic theory of Anosov diffeomorphisms*, pp. 61–73. Springer, 2008.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, et al. Jax: Autograd and xla. *Astrophysics Source Code Library*, pp. ascl–2111, 2021.

Will Dabney, Mark Rowland, Marc Bellemare, and Rémi Munos. Distributional reinforcement learning with quantile regression. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI'18)*, 2018.

Deniz Eroglu, Jeroen SW Lamb, and Tiago Pereira. Synchronisation of chaos and its applications. *Contemporary Physics*, 58(3):207–243, 2017.

Marc Finzi, Shikai Qiu, Yiding Jiang, Pavel Izmailov, J Zico Kolter, and Andrew Gordon Wilson. From entropy to epiplexity: Rethinking information for computationally bounded intelligence. *arXiv preprint arXiv:2601.03220*, 2026.

Sabino Gadaleta and Gerhard Dangelmayr. Optimal chaos control through reinforcement learning. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 9(3):775–788, September 1999.

Lukas Halekotte, Anna Vanselow, and Ulrike Feudel. Transient chaos enforces uncertainty in the british power grid. *Journal of Physics: Complexity*, 2(3):035015, 2021.

Kensuke Ikeda. Multiple-valued stationary state and its instability of the transmitted light by a ring cavity system. *Optics Communications*, 30(2):257–261, 1979.

Ruoxi Jiang, Peter Y. Lu, Elena Orlova, and Rebecca Willett. Training neural operators to preserve invariant measures of chaotic attractors. *Advances in Neural Information Processing Systems*, December 2023.

Edward N. Lorenz. Predictability: Does the flap of a butterfly's wings in Brazil set off a tornado in Texas? In *Proceedings of the 139th Meeting of the American Association for the Advancement of Science*, 1972.

Benoit B Mandelbrot. *Fractals and Chaos: the Mandelbrot Set and Beyond*. Springer, 2004.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing Atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

Ricard Montalà, Bernat Font, Pol Suárez, Jean Rabault, Oriol Lehmkuhl, Ricardo Vinuesa, and Ivette Rodríguez. Deep Reinforcement Learning for Active Flow Control Around a Three-dimensional Flow-separated Wing at Re = 1,000. *arXiv preprint arXiv:2509.10195*, 2025.

Shijie Qin and Shijun Liao. A kind of Lagrangian chaotic property of the Arnold–Beltrami–Childress flow. *Journal of Fluid Mechanics*, 960:A15, 2023.

Gerasimos Rigatos, Pierluigi Siano, and Taniya Ghosh. A nonlinear optimal control approach to stabilization of business cycles of finance agents. *Computational Economics*, 53(3):1111–1131, 2019.

Mark Rowland, Marc Bellemare, Will Dabney, Rémi Munos, and Yee Whye Teh. An analysis of categorical distributional reinforcement learning. In *Proceedings of the 21th International Conference on Artificial Intelligence and Statistics (AISTAT'18)*, pp. 29–37. PMLR, 2018.

James Rudd-Jones, Mirco Musolesi, and María Pérez-Ortiz. Multi-agent reinforcement learning simulation for environmental policy synthesis. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'25)*, 2025a.

James Rudd-Jones, Fiona Thendean, and María Pérez-Ortiz. Crafting desirable climate trajectories with reinforcement learning explored socio-environmental simulations. *Environmental Data Science*, 4:e41, 2025b.

James Rudd-Jones, María Pérez-Ortiz, and Mirco Musolesi. Understanding individual decision-making in multi-agent reinforcement learning: A dynamical systems approach. In *Proceedings of the 3rd Conference on Reinforcement Learning (RLC'26)*, 2026.

David Ruelle. A measure associated with axiom-a attractors. *American Journal of Mathematics*, pp. 619–654, 1976.

Yuzuru Sato, Eizo Akiyama, and J Doyne Farmer. Chaos in learning a simple two-person game. *Proceedings of the National Academy of Sciences*, 99(7):4748–4751, 2002.

Yair Schiff, Zhong Yi Wan, Jeffrey B. Parker, Stephan Hoyer, Volodymyr Kuleshov, Fei Sha, and Leonardo Zepeda-Núñez. DySLIM: Dynamics Stable Learning by Invariant Measure for Chaotic Systems. In *Proceedings of the 41st International Conference on Machine Learning (ICML'24)*, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Ippei Shimada and Tomomasa Nagashima. A numerical approach to ergodic problem of dissipative dynamical systems. *Progress of Theoretical Physics*, 61(6):1605–1616, 1979.

Yakov G Sinai. Gibbs measures in ergodic theory. *Russian Mathematical Surveys*, 27(4):21, 1972.

Steven H Strogatz. *Nonlinear Dynamics and Chaos: with Applications to Physics, Biology, Chemistry, and Engineering*. Chapman and Hall/CRC, 2024.

KGD Sulalitha Priyankara, Sanjeeva Balasuriya, and Erik Bollt. Quantifying the role of folding in nonautonomous flows: The unsteady double-gyre. *International Journal of Bifurcation and Chaos*, 27(10):1750156, 2017.

Ke Sun, Bei Jiang, and Linglong Kong. How does return distribution in distributional reinforcement learning help optimization? In *Workshop on Aligning Reinforcement Learning Experimentalists and Theorists (ARLET'24)*, 2022.

Kristen M Thyng, Chad A Greene, Robert D Hetland, Heather M Zimmerle, and Steven F DiMarco. True colors of oceanography: Guidelines for effective and accurate colormap selection. *Oceanography*, 29(3):9–13, 2016.

Marcelo Viana. *Stochastic Dynamics of Deterministic Systems*, volume 21. IMPA, 1997.

Tao Wang, Sylvia Herbert, and Sicun Gao. Fractal Landscapes in Policy Optimization. *Advances in Neural Information Processing Systems*, 2023.

Tao Wang, Sylvia Lee Herbert, and Sicun Gao. Mollification Effects of Policy Gradient Methods. In *Proceedings of the 41st International Conference on Machine Learning (ICML'24)*, 2024.

Alan Wolf, Jack B Swift, Harry L Swinney, and John A Vastano. Determining Lyapunov exponents from a time series. *Physica D: Nonlinear Phenomena*, 16(3):285–317, 1985.

Rory Young and Nicolas Pugeault. Enhancing robustness in deep reinforcement learning: A lyapunov exponent approach. *Advances in Neural Information Processing Systems*, 37:86102–86123, 2024.

A Experimental Code

Code for all the experiments will be released upon acceptance. The chaotic control environments are part of a github package that will be linked here upon acceptance to ensure double blind is respected. We'd like to thank Thyng et al. (2016) for creating `cmocean`, a great source of colour maps used throughout this paper!

B Proof of Theorem 4.5

Theorem (Exponential one-step sensitivity of scalar sample returns)**.** *Under Assumptions 4.1 and 4.2, for any fixed action sequence, the truncated return map $s \mapsto G_T(s)$ is Lipschitz (*$\mathrm{Lip}(G_T)$*) with constant*

$$\mathrm{Lip}(G_T) \;\le\; K_R \sum_{t=0}^{T-1} (\gamma K_f)^t \;=\; K_R \cdot \frac{(\gamma K_f)^T - 1}{\gamma K_f - 1} \quad (\gamma K_f \neq 1). \tag{16}$$

In particular, if $\gamma K_f > 1$, then $\mathrm{Lip}(G_T)$ grows exponentially in T.

Proof. Let s_t and $\tilde{s}_t$ be trajectories that start from s_0 and $\tilde{s}_0$ under the same closed-loop policy π. By the closed-loop dynamics Lipschitz property, $\|s_t - \tilde{s}_t\| \leq K_f^t \|s_0 - \tilde{s}_0\|$ by induction. Then:

$$|G_T(s_0) - G_T(\tilde{s}_0)| \leq \sum_{t=0}^{T-1} \gamma^t |R(s_t, \pi(s_t)) - R(\tilde{s}_t, \pi(\tilde{s}_t))| \leq \sum_{t=0}^{T-1} \gamma^t K_R \|s_t - \tilde{s}_t\| \leq K_R \sum_{t=0}^{T-1} (\gamma K_f)^t \|s_0 - \tilde{s}_0\|. \tag{17}$$

□

C Proof of Theorem 4.6

Theorem (W_1-smoothness of the return distribution under statistical stability)**.** *Under Assumptions 4.2 - 4.4, for any fixed initial action a the following holds:*

$$W_1(Z^\pi(s_1, a), Z^\pi(s_2, a)) \leq \frac{K_R}{1 - \gamma K_P} \|s_1 - s_2\|. \tag{18}$$

Proof. Let $\mathcal{T}^\pi$ be the distributional Bellman operator: $(\mathcal{T}^\pi Z)(s, a) \overset{D}{=} R(s, a) + \gamma Z(S', A')$ with $S' \sim P(\cdot \mid s, a)$ and $A' \sim \pi(\cdot \mid S')$. Let us fix a and define:

$$K := \sup_{s_1 \neq s_2} \frac{W_1(Z^\pi(s_1, a), Z^\pi(s_2, a))}{\|s_1 - s_2\|}. \tag{19}$$

Using standard coupling plus the fact that W_1 contracts under γ-scaling,

$$W_1(Z^\pi(s_1, a), Z^\pi(s_2, a)) \leq |R(s_1, a) - R(s_2, a)| + \gamma W_1\left(Z^\pi(S_1', A_1'), Z^\pi(S_2', A_2')\right). \tag{20}$$

Now couple S_1', S_2' using an optimal W_1 coupling for $P(\cdot \mid s_1, a)$ and $P(\cdot \mid s_2, a)$, so $\mathbb{E}\|S_1' - S_2'\| \leq K_P \|s_1 - s_2\|$. Conditioning on (S_1', S_2') and using that $Z^\pi(\cdot, a)$ is K-Lipschitz with respect to W_1 (by the definition of K), the mixture stability property of W_1 implies:

$$W_1\left(Z^\pi(S_1', A_1'), Z^\pi(S_2', A_2')\right) \leq \mathbb{E}\big[W_1(Z^\pi(S_1', A_1'), Z^\pi(S_2', A_2'))\big] \leq K \mathbb{E}\|S_1' - S_2'\|. \tag{21}$$

Here we use the standard inequality that W_1 between mixtures is bounded by the expectation of W_1 under a coupling of the mixing variables. Assumption 4.2 ensures all terms are well defined. Therefore,

$$W_1(Z^\pi(s_1, a), Z^\pi(s_2, a)) \leq K_R \|s_1 - s_2\| + \gamma K K_P \|s_1 - s_2\|. \tag{22}$$

Divide by $\|s_1 - s_2\|$ and take the Supremum:

$$K \leq K_R + \gamma K K_P, \tag{23}$$

hence $K \leq \frac{K_R}{1 - \gamma K_P}$ when $\gamma K_P < 1$. Thus concluding the proof.

□

D When Statistical Stability Fails

We now consider the regime in which Assumption 4.3 fails, and clarify how distributional RL behaves in this setting in more detail.

Failure of Statistical Stability. Assumption 4.3 requires that the transition kernel varies Lipschitz continuously in W_1 with respect to the state. This can fail at bifurcation points or near basin boundaries. Consider a system with two disjoint attractors separated by a stable manifold. Two arbitrarily close states on opposite sides of the boundary evolve toward different attractors:

$$P(\cdot \mid s_1, a) = \delta_{\mathcal{A}_L}, \quad P(\cdot \mid s_2, a) = \delta_{\mathcal{A}_R}, \tag{24}$$

where $\delta_{\mathcal{A}_R}$ defines a Dirac measure of the "right" attractor, and conversely $\delta_{\mathcal{A}_R}$ a "left" attractor. Then

$$W_1(P(\cdot \mid s_1, a), P(\cdot \mid s_2, a)) = d(\mathcal{A}_L, \mathcal{A}_R), \tag{25}$$

which stays bounded away from zero and does not reduce even as $\|s_1 - s_2\| \to 0$, where $d(\mathcal{A}_L, \mathcal{A}_R)$ denotes the distance between the two attractors in the underlying state space metric. In this case

K_P is unbounded and Theorem 4.6 no longer applies. Such behaviour occurs at tipping points, bifurcations, or in perfectly deterministic systems without noise.

Expectation Based Learning Near Basin Boundaries. When basin switching occurs, the scalar value function:

$$V^\pi(s) = \mathbb{E}[Z^\pi \mid s_0 = s], \tag{26}$$

can behave like a step function across the boundary. Small perturbations in state produce large jumps in the bootstrapped target, leading to large TD errors, unstable gradients, high sensitivity to initialisation and step size. Moreover, expectation collapses multimodal outcomes. If one branch yields $+100$ and the other -100, the learned value is 0, which may correspond to no realisable trajectory of the system.

How Distributional RL Fares. Even when Assumption 4.3 fails, distributional RL still preserves multimodality. Near a separation point the return distribution is naturally bimodal, and distributional methods represent both modes rather than collapsing them into a single mean. Further, the distributional Bellman operator involves only translation by $R(s,a)$ and scaling by γ:

$$Z(s,a) \overset{D}{=} R(s,a) + \gamma Z(S', A'). \tag{27}$$

Both operations are non-expansive in Wasserstein distance: translation leaves the distance unchanged, while scaling by γ contracts it:

$$W_1(X + c, Y + c) = W_1(X, Y), \quad W_1(\gamma X, \gamma Y) = \gamma W_1(X, Y). \tag{28}$$

Thus, discounting remains contractive even when transitions are discontinuous. There are therefore two important regimes. If $\gamma K_P < 1$, Theorem 4.6 guarantees Lipschitz smoothness of the return distribution in W_1. If statistical stability fails (e.g. at bifurcations), this guarantee disappears. However, distributional RL still avoids expectation collapse and provides a better conditioned loss geometry than scalar TD learning.

E Ikeda Map One-Step Objective Figure

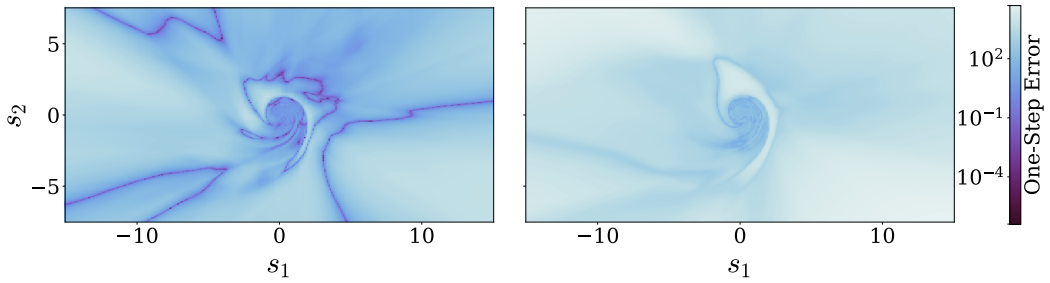

Figure 6: Ikeda Map objective landscapes. Left: DQN, Right: QRDQN. Figure data collected from agents after $3e8$ training steps.

F QRDQN Distribution for Logistic Map

To analyse the structure learned by QRDQN in the $1D$ environment, we visualise the full return distribution as a function of state. For a fixed (or greedy) action, the network outputs N quantile locations $\{z_i(s)\}_{i=1}^N$ at quantile fractions $\tau_i = \frac{i+0.5}{N}$, which approximate the inverse cumulative distribution function (CDF) of the return $Z(s)$. From these quantile points we construct (i) an empirical CDF surface:

$$F(z \mid s) \approx \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}[z_i(s) \leq z], \tag{29}$$

and (ii) an approximate probability density function (PDF) surface obtained by histogramming the quantile locations at each state and normalising to unit mass.

Figure 7 visualises the return distribution learned by QRDQN for the Logistic Map as a function of state. In the PDF surface, probability mass concentrates in narrow ridges that shift smoothly across s,

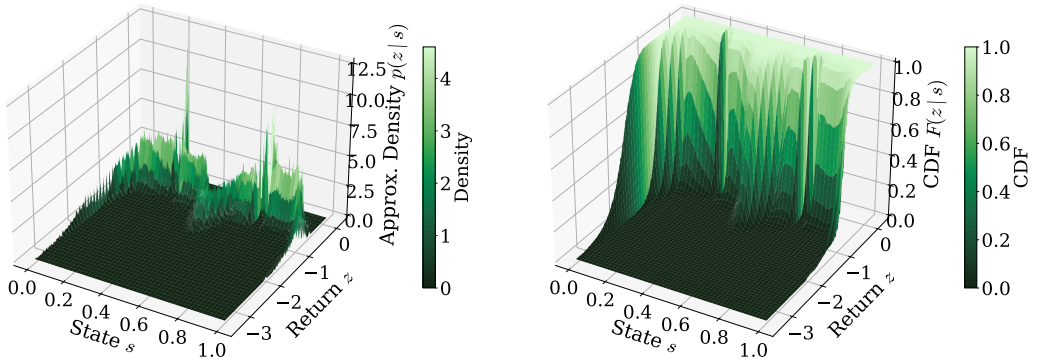

Figure 7: Left: QRDQN implied return PDF surface. Right: QRDQN implied return CDF surface.

indicating that only a small number of trajectory outcomes dominate for each state while their relative likelihood evolves gradually. Regions with multiple peaks correspond to states where trajectories can follow qualitatively different regimes, producing multi-modal returns. The CDF surface shows steep transitions where many quantiles collapse around similar return values, revealing discrete bands of likely outcomes. Although these transitions are sharp in the return dimension, their location varies smoothly across state. Together, these patterns show that while individual trajectories diverge under chaotic dynamics, the return distribution evolves coherently across state, supporting our claim that modelling the full distribution captures structured variability that expectation based methods would collapse into a single value.

G Implementation Details

In this section, we outline the exact hyperparameters and network architectures steps used for our experiments to ensure full reproducibility. All algorithms were implemented using Jax (Bradbury et al., 2021).

G.1 Network Architectures

Value-based methods (DQN, QRDQN): We use the same network for both. The network consists of five hidden dense layers (feature sizes: $64, 128, 256, 128, 64$). All hidden layers utilise ReLU activation functions. For DQN, the output layer consists of a single linear layer matching the action space $|\mathcal{A}|$. For QRDQN, the output layer dimension is $|\mathcal{A}| \times N$, where N is the number of quantiles, representing the distribution of returns for each action.

Actor-Critic method (PPO): The actor and critic networks share the same hidden layer base consisting of two hidden dense layers (feature sizes: $256, 256$) but branch into separate fully connected heads. The policy head outputs a Gaussian distribution over actions, while the value head outputs a single scalar baseline.

G.2 Hyperparameters

Table 1 summarises the hyperparameters used for DQN, QRDQN, and PPO.

Table 1: Hyperparameter configurations for DQN, QRDQN, and PPO.

Algorithm	Hyperparameter	Value
Common / DQN	Optimiser	Adam
	Learning rate	1×10^{-3}
	Discount factor (γ)	0.99
	Replay buffer size	1×10^{7}
	Batch size	64
	Target network update frequency	64 steps
	Initial exploration ϵ	1.0
	Final exploration ϵ	0.01
	Exploration ϵ decay steps	1×10^{6}
QRDQN	Number of quantiles (N)	201
	Huber loss threshold (κ)	1.0
PPO	Optimiser	Adam
	Learning rate	3×10^{-4}
	Number of parallel environments	32
	Rollout length per environment	256
	PPO clip parameter (ϵ)	0.2
	GAE parameter (λ)	0.95
	Value loss coefficient	0.5
	Entropy coefficient	0.01
	Optimisation epochs	4