On the Evolution of Return Distributions in Continuous-Time Reinforcement Learning

Harley Wiltzer
School of Computer Science
McGill University, Montreal
November 2021

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of Master of Science

Abstract

This thesis develops the theory of distributional reinforcement learning in the continuous-time setting. Inspired by the literature on continuous-time reinforcement learning and optimal control, we demonstrate that existing (discrete-time) distributional reinforcement learning algorithms may fail to converge on the correct return distributions even in very simple environments. To account for this, we characterize the return distributions induced by a broad class of continuous-time stochastic Markov Reward Processes, and we use this characterization to inform distributional reinforcement learning algorithms to account for continuous-time evolution. The characterization takes the form of a family of partial differential equations on the space of return distributions. Furthermore, we address the issue of the representation of arbitrary probability measures with bounded space, and in doing so we show how under a particular choice of representation, the return distributions are characterized by a set of Hamilton-Jacobi-Bellman equations, which are ubiquitous in the optimal control literature. We then demonstrate a construction of a continuous-time distributional algorithm and study its convergence properties in the policy evaluation setting. Finally, we provide an implementation using deep neural networks and evaluate its performance empirically against various benchmarks.

Abrégé

Cette thèse développe la théorie de l'apprentissage par renforcement au niveau des distributions de probabilité des gains, dans le cas où le temps est continu. Inspirés par la littérature du théorie de côntrole optimal et simplement l'apprentissage par renforcement avec l'évolution de temps continu, nous démontrons que les algorithmes existants ne peuvent pas toujours parvenir à converger sur les distributions de gains propres, même lorsque l'environment est très simple. Pour rendre compte de ce résultat, nous caractérisons les distributions des gains provoquées par une classe vaste de processus Markoviens de récompenses stochastiques. Cette caractérisation, qui prend la forme d'une famille d'équations differentielles aux dérivées partielles, est ensuite utilisée pour informer la conception d'algorithmes d'apprentissage par renforcement qui modèlent les distributions de gains qui évoluent continuellement. De plus, nous addressons le problème de la representation des distributions probabilistes avec l'espace limité, et nous montrons common choisir la representation pour transformer la caractérisation des distributions de gains à un ensemble d'équations Hamilton-Jacobi-Bellman, qui sont omniprésentes dans la littérature du contrôle optimal. Nous construisons une algorithme pour apprendre ces representations en temps continu, et nous étudions ses propriétés concernant la convergance vers des distributions de gains stationnaires en cas de l'évaluation des stratégies. Finalement, nous fournissons une implémentation de cette algorithme avec des résaux profonds, et nous validons ses performances empiriquement contre plusieurs points de référence.

Acknowledgements

I am immensely thankful to my supervisors Dave Meger and Marc Bellemare for their perpetual mentorship and guidance over the course of my Masters'. Over the past year and a half of thesis work, I have repeatedly been impressed by the patience they had for the many rabbit holes I got lost in, the lulls in progress, and the occasional technical issues I've had due to my stubborn computer preferences. I truly commend them for not giving up on me in the many moments where I felt like my efforts were futile.

Thanks as well to Dave Meger, Greg Dudek, and Hsiu-Chin Lin for making the Mobile Robotics Lab at McGill such an amazing environment to be a part of, as well as the many MRL students that have always made me feel welcome. Special thanks to Jean-François Tremblay, Wesley Chung, Melissa Mozifian, Sahand Rezaei Shostari, Lucas Berry, Andrew Holliday, and Abhisek Konar for the many sessions of discussion and guidance along the way. I hope to see you all in person one day.

Marc Bellemare's groups at Mila and Google have also been a great source of inspiration. I am very thankful for the several discussions with Ross Goroshin about fancy techniques in continuous-time optimal control, which have contributed a lot to my appreciation of the field.

Additionally, I must thank Professors Prakash Panangaden, Luc Devroye and Siamak Ravanbahksh for the invaluable challenges and assistance along the way, which sincerely enriched my appreciation for mathematics. This thesis would look very different without your influence.

I am extremely indebted to the wonderful community of free[1] open source software developers that deserve tons of credit for the role they've played in improving my productivity. In particular, I thank the Gentoo developers, especially those that have directly assisted me numerous times on IRC, Matthew Johnson and all the other Jax developers for fixing issues unbelievably quickly, and Linus for obvious reasons.

Lastly, I thank my family and friends for their encouragement over the years. In particular, I am very grateful to my parents, Casey, and my grandparents for their never-ending support.

[1] As in freedom, of course, rather than beer

Contents

List of Figures

List of Algorithms

"... on the planet Earth, man had always assumed that he was more intelligent than dolphins because he had achieved so much – the wheel, New York, wars and so on – whilst all the dolphins had ever done was muck about in the water having a good time. But conversely, the dolphins had always believed that they were far more intelligent than man – for precisely the same reasons."

Douglas Adams, *The Hitchhiker's Guide to the Galaxy*

1 Introduction

Reinforcement learning (RL) is a form of artificial intelligence with the ambition of creating *general purpose* problem-solving algorithms that improve with experience. Unlike other machine learning tasks, generally RL algorithms begin with no understanding of the problem to be solved, and are not given any data to learn from. Consequently, aside from learning how to solve a problem, an RL algorithm must also learn how to gather data to improve itself by maximizing the long term *returns* accumulated by the agent due to good behavior. A common paradigm in RL is based on estimating the expected value, measured in cumulative future rewards, should the agent follow a given strategy. Due to the uncertainty of how the agent's actions affect the environment and the rewards, estimating the expected value of a strategy can be quite difficult.

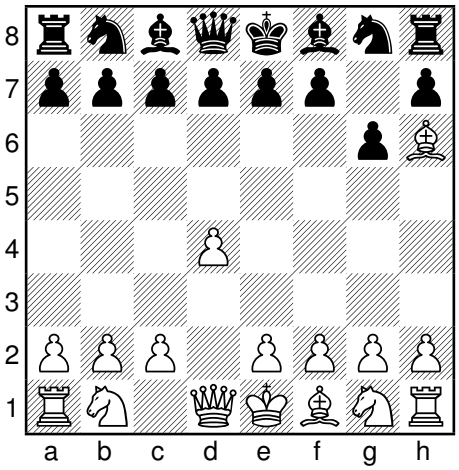
The expectation of the return, however, is not necessarily the best metric for evaluating strategies. Expected values are most meaningful when the random variable can be sampled arbitrarily many times, in which case the many samples "balance each other out" to a net quantity that is approximated well by the expectation. However, this is not always (and perhaps not even usually) the setting that RL algorithms find itself in. When fewer samples can be drawn, individual samples have a much larger impact and may never be "balanced out".

More concretely, consider a scenario in which an agent is presented with a wager, and the agent can decide whether to take the wager or not. Suppose the wager costs \$1,000, and by playing the wager the agent wins \$100,000 with probability $1/10$. The expected value of this wager is simply calculated as $(1/10) \times \$100000 - (9/10) \times \$1000 = \$9,100$. Using expected value as a means of decision making, we see that the agent should take the wager, as it expects a profit of \$9,100 each time the wager is played. However, often this kind of reasoning can fail, especially when the agent has no knowledge of how the rewards are generated. Reinforcement learning agents are generally not assumed to have any such knowledge, so they have to estimate it by observing samples of state transitions and rewards from the environment (or more commonly, a simulator of the environment). Suppose the agent observes many samples and is very confident in its estimate of the expected return for the wager, and is subsequently given only three more opportunities to play. More likely than not, the agent will not win the wager within three attempts. Should the agent still play the wager since its expected value is high, or should it simply pass since most likely it'll lose $\$3,000$? At the very least, one can make a reasonable argument for each choice. In particular, if the agent would not have enough money to feed its children if it were to lose $\$3,000$, it is likely that most people would agree that playing the wager is irresponsible, and ultimately the "correct" decision depends on the agent's ethos.

Interestingly, there are relatively common scenarios where the more dangerous scenario might be preferred by some people when an experiment cannot be run as many times as desired. An amusing example of this is popular in online speed chess, where players have very little time to spend pondering moves. The following is a demonstration of *the Lefong trap*[1] that is sometimes played in these games:

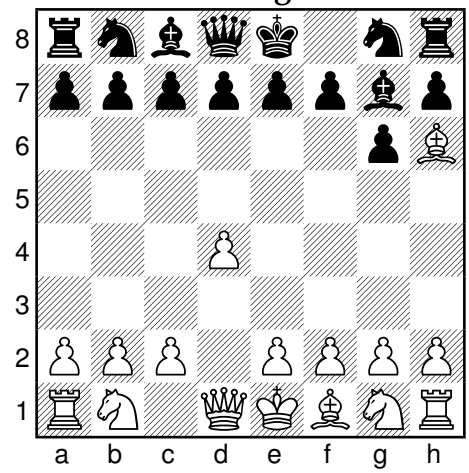
1 d4 g6 2 ♗h6

[1]This trap was popularized by the Canadian FIDE Master Lefong Hua.

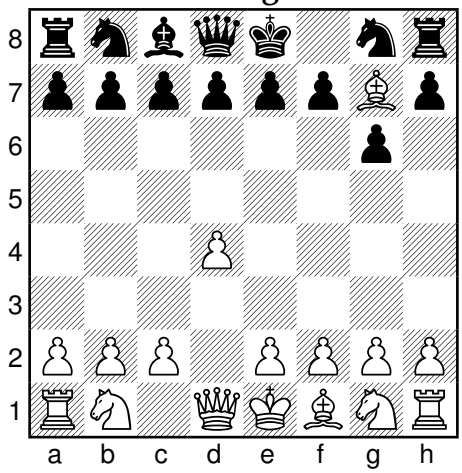


White's second move, with regard to general chess strategy, is horrendous: the move leaves the bishop undefended and in the line of attack of black's bishop and knight. White was likely hoping for the following continuation:

2. . . ♗g7



3 ♗×g7



Black's second move in this hypothesized continuation from white is even worse than white's second move! One may reasonably wonder then why the Lefong trap is ever played, and the reason is simple: in such fast chess games, sites allow the players to "pre-move" – that is, commit to a move during the opponent's turn, to avoid spending any time on their own turn. When black played the move `g6`, their intention was almost surely to follow it with `Bg7` (this is called *the modern defense*), making it a great candidate for a pre-move. White exploits this by playing a horrible (but unaccounted for) move that only works because black, hopefully, waives his ability to respond to `Bh6`. If black *does not* pre-move `Bg7`, white's `Bh6` loses them the game. In 2018, teenaged grandmaster Andrew Tang defeated Magnus Carlsen, the world champion and highest rated player of all time, using the Lefong trap[2].

[2]See https://www.youtube.com/watch?v=Kr5sxSja2D8.

The expected return of the Lefong move `Bh6` is surely far from optimal. Should Andrew Tang have attempted this move game after game, it would fail far more often than not, so his strategy in this case could not have been based on the expected value of his move. However, given that the move would not be played many times, and he may never have the opportunity to beat a world champion with the Lefong again, Andrew Tang was able to justify his move.

The theory of *distributional* RL can aid in addressing these types of conundrums by learning the entire probability distribution over the cumulative future returns due to a given strategy, as opposed to just the expected value. Given an understanding of the distribution over returns, one has much more information at their disposal to aid in decision-making, for example, by accouting for the variance of the return to make the risk-averse decision of declining a wager, or by preferring decisions that lead potential to exceptionally high rewards like defeating a world champion at their game.

Since its introduction in Bellemare et al. [2017a], distributional RL has gained lots of interest within the reinforcement learning community, partly because of its impressive empirical performance. Even when distributional RL is employed and decisions are made just by comparing the means of return distributions, distributional RL still tends to outperform its expected value counterparts. Bellemare et al. [2017a] attributes this to the fact that by modeling potential multimodalities in the return distributions, distributional algorithms may be less sensitive to noise in stochastic training procedures. Additionally, reinforcement learning algorithms tend to approximate returns under the assumption that the policy is not changing over time, which generally is not the case – of course, in order for an agent to improve at a task, it must change its policy. By modeling the full distribution over returns, this phenomenon can be manifested in the uncertainty associated with return distributions, which is believed to help stabilize training.

Moreover, another interesting prospect for learning return distributions is that they can be used to promote *exploration* in a principled manner. Since RL algorithms usually have to collect their own data in order to learn, it is never really clear to them if their current idea of an optimal strategy *is* in fact optimal, unless they are able to try every strategy in every possible scenario. This is generally impossible. Despite being studied since the birth of RL research, exploration still remains a major challenge, as well as a principle contributor to the poor sample complexity often observed in reinforcement learning. Given estimates of return distributions, however, it may be possible to use uncertainty in the return as a proxy for determining which strategies to learn more about [Mavrin et al., 2019].

A long-standing issue in reinforcement learning research is that the literature usually studies systems that evolve in discrete, fixed-duration timesteps. Of course, the real world does not work this way, and even many of the synthetic benchmarks are actually modeling processes that evolve continuously in time. Not accounting for continuous-time processes in RL can lead to detriments in training time, their ability to correctly model the value function, and performance [Doya, 2000, Munos, 2004, Tallec et al., 2019].

The analysis of continuous-time processes, however, incurs substantial mathematical challenges that are not present in discrete time. Even for fully deterministic processes with very smooth dynamics and simple controls, in general the value function cannot be characterized in a "classical", intuitive sense. This is because in the continuous-time limit, the value function does not preserve enough "smoothness", so it must instead be interpreted as a weakened notion of a solution to a PDE [Crandall and Lions, 1983]. Existing work in continuous-time reinforcement learning and optimal control has addressed stochasticity in the dynamics and the policy, but refrains from studying the distribution of the random return by estimating only its mean. The principal goal of this thesis is to explore the behavior of the return distribution function in the continuous time limit. Given the undesirable non-smoothness of the value function in continuous time, it is only natural to suspect that the return distribution function, being a function into an infinite-dimensional space of probability measures, will have a non-trivial characterization (if it exists at all). We will show that indeed the return distribution function does exist, and its uniqueness can be established in a weak sense.

Beyond the analytical understanding, we must consider the computational challenges involved in estimating the return distribution function, whose image is infinite-dimensional. We will show that the manner in which probability measures are represented will be reflected in the PDE governing the evolution of the return distributions, which is a consequence that has no equivalent manifestation in discrete time. We will also discuss a class of representations of probability measures that induce a simple and familiar form of the characterization, and use this knowledge to study computationally-tractable algorithms for distributional policy evaluation that is convergent in the continuous time limit.

Aside from the concurrent work of Halperin [2021], to our knowledge, distributional RL has not been studied in the continuous-time setting. This thesis will substantially broaden the theory of continuous time distributional reinforcement learning by analyzing the characteristics of the evolution of return distributions, providing tractable reinforcement learning algorithms that learn return distributions that are convergent in the continuous-time limit, and by demonstrating that some of the problems with learning

value functions in continuous-time RL are exacerbated when estimating return distributions in continuous-time.

The thesis will be organized as follows. Chapter 2 provides an overview of the literature of reinforcement learning and related fields, and discusses some important results that will be useful in the development moving forward. Next, in Chapter 3, we study how return distributions evolve in time and ultimately derive a partial differential equation that characterizes return distributions induced by a vast class of stochastic processes. Chapter 4 is concerned with framing continuous-time distributional RL as an optimization in the space of probability measures, as well as methods of representing probability measures and continuous-time evolutions in a tractable manner. In Chapter 5, we present the DEICIDE framework for the construction of continuous-time distributional reinforcement leanring algorithms, and we outline a selection of algorithm examples. Empirical results of these algorithms are given in §5.2.

2

Background

This section will give a concise background of the mathematical concepts that will be useful in the sequel, as well as a review of the literature that this work builds on. It is assumed that the reader is familiar with multivariable calculus, linear algebra, and the analysis of algorithms. Section §2 gives an overview of the notational conventions used throughout the remainder of the thesis, and the remaining sections briefly cover the main results leading up to my research.

Notation

Symbol	Meaning
$\mathbf{R}$	The set of real numbers
$\mathbf{R}_+$	The set of nonnegative real numbers
$\mathbf{N}$	The set of natural numbers, $\{1, 2, \dots\}$
$\mathbf{N}_0$	The set of natural numbers including zero, $\mathbf{N} \cup \{0\}$
2^X	The powerset (set of all subsets) of a set X
Set	The category with sets as objects and functions as morphisms
$[N]$	When N is an integer, $[N] \triangleq \{1, \dots, N\}$.

$\mathscr{B}(\mathcal{A})$	The Borel σ-algebra of the topological space $\mathcal{A}$.
$\mathsf{Ran}(f)$	The range of a function f
$\mathbf{E}\left[f(X)\right]$	The expected value of $f(X)$, where X is a random variable
$\mathbf{E}\left[f(X) \mid Y\right]$	The conditional expectation of $f(X)$ given the value of a random variable Y
$X \overset{\mathcal{L}}{=} Y$	Equality in law: the random variables X, Y are distributed identically
$\mathcal{H}(p)$	The (differential) entropy of a probability distribution p
$\mathcal{H}(p, q)$	The (differential) cross-entropy between probability distributions p and q, defined as $\mathcal{H}(p, q) = -\mathbf{E}_p[\log q]$
$\mathsf{U}\left(X\right)$	The uniform distribution over a bounded set X
$C(A; B)$	The set of continuous functions $f : A \to B$, endowed with the supremum norm
$C(A)$	Equivalent to $C(A; B)$ when the output space B is unambiguous
$C^k(A; B)$	The subset of $C(A; B)$ with functions having continuous kth derivatives, for $k \in \mathbf{N}$
$C^\infty(A; B)$	The subset of continuously differentiable functions in $C(A; B)$
$C_c^k(A; B)$	The subset of $C^k(A; B)$ containing functions that are supported on a precompact set, $k \in \mathbf{N} \cup \{\infty\}$
$C_0^k(A; B)$	The set of functions f in $C_c(A; B)$ that are 0 on the boundary of the support of f (Dirichlet boundary conditions)
$\mathsf{AC}(A)$	The set of absolutely continuous functions over a set A
$\mathcal{P}_p\left(\mathcal{X}\right)$	The set of probability measures over a measurable space $\mathcal{X}$
$\mathbf{W}_p\left(\mathcal{X}\right)$	The set of probability measures over a set $\mathcal{X}$ endowed with the p-Wasserstein metric
$a \wedge b$	The minimum of a and b
$a \vee b$	The maximum of a and b
$\langle f, g \rangle$	The inner product between elements f, g of an arbitrary inner product space
$\langle f, g \rangle_{\mathcal{S}}$	The inner product between vectors f, g in an inner product space $\mathcal{S}$
$\otimes$	Pointwise (tensor) product: $f \otimes g = x \mapsto f(x)g(x)$
$\mathbf{1}_{[\mathsf{predicate}]}$	The function that takes the value 1 when predicate is true, and 0 otherwise
χ_A	The characteristic function of a measurable set A: $\chi_A(x) = \mathbf{1}_{[x \in A]}$

δ_z	The Dirac delta distribution[1]. This is defined such that for any function f, $\int \delta_z(x) f(x) dx = f(z)$.
id	The identity function, $\mathsf{id}(x) = x$
Tr	The trace operator
$\mathsf{J_x}$	The Jacobian operator for vector-valued functions with respect to variable(s) $\mathbf{x}$
$\mathsf{H_x}$	The Hessian operator with respect to variable(s) $\mathbf{x}$
$\frac{\delta F}{\delta u}$	The first variation of a functional F with respect to its parameter u
$\mathscr{D}(\mathscr{L})$	The domain of an operator $\mathscr{L}$
$\perp(\cdot)$	The "stop gradient operator". It satisfies $\perp(f)(x) = f(x)$, but $\nabla \perp(f)(x) \equiv 0$
ι_k	The projection map to dimension k; If $x = (x_1, \ldots, x_k, \ldots x_n)$, then $\iota_k x = x_k$.

2.1 Reinforcement Learning

The goal of Reinforcement Learning (RL) is to study how an *agent* can develop a behavioral policy that is expected to successfully perform an abstract task, where success is measured by the *reward* it receives by interacting with the environment. As an example, we can think of the agent as a humanoid robot that is rewarded for each second it is balancing on one leg, and penalized (say, by receiving a negative reward) for falling down. A robot that receives a high reward in this example will have demonstrated an ability to balance itself on one foot.

Abstractly, in RL, an agent repeatedly undergoes the following cycle,

1. The agent observes the present state from the environment;
2. Based on the present state, the agent performs an action of its choice;
3. The agent subsequently transitions to a new state and receives a reward.

The goal of the agent is to maximize the total reward that it accumulates. Such a setting is formally described by a *Markov Decision Process*.

Definition 1 (Markov Decision Process, [Puterman, 2014])**.** A *Markov Decision Process* (MDP) is a 5-tuple $(\mathcal{X}, \mathcal{A}, \mathcal{R}, P, \gamma)$, where

[1]Note that "distribution" in this context refers to a generalized function, and not a probability distribution.

1. $\mathcal{X}$ is a set, called the *state space*, whose elements are referred to as *states*;
2. $\mathcal{A}$ is a set, called the *action space*, whose elements are referred to as *actions*;
3. $r : \mathcal{X} \to \mathbf{R}$ is the *reward function*;
4. $P : \mathcal{X} \times \mathcal{A} \to \mathscr{P}(\mathcal{X})$ is called the *Markov kernel*, where $P(x' \mid x, a)$ denotes the probability of the agent transitioning from state $x \in \mathcal{X}$ to state $x' \in \mathcal{X}$ upon performing action $a \in \mathcal{X}$;
5. $\gamma \in (0, 1)$ is the *discount factor*, which serves the purpose of discounting the value of future rewards.

$\triangledown$

We note that it is common in discrete-time RL to model "stochastic reward functions", in which case the reward function r is replaced with an "augmented" Markov kernel $P^\dagger(x', r \mid x, a)$, which models the probability density of the next state and reward given the current state and action. Furthermore, occasionally the return is modeled as a function of both the current state and action – note however that the same effect can be simulated roughly by augmenting each state with the last executed action. In this thesis, we will only consider deterministic reward functions over the state space.

It will often be convenient to refer to the probability measure $P^\pi \in \mathcal{P}_p(X \times X)$, given by

$$P^\pi(x' \mid x) = \int_{\mathcal{A}} P(x' \mid x, a)\pi(da \mid x) \tag{2.1}$$

where $\pi : \mathcal{X} \to \mathcal{P}_p(\mathcal{A})$ is a *policy* that samples actions from a state-conditioned distribution.

An important remark about MDPs is that they satisfy the *Markov property*, that is, rewards and state transitions are statistically independent from all states and actions from the past, and depend only on the present state and action.

Despite the simplicity of the MDP model, the space of problems that can be formulated as MDPs is enormous [Barto et al., 1981]. In fact, it has been hypothesized by leading researchers that RL agents can achieve *artificial general intelligence* [Silver et al., 2021], implying that they can learn anything that a human can. While RL cannot (yet) achieve human-level intelligence, RL has successfully achieved superhuman performance in complex board games like backgammon [Tesauro, 1994], chess, Shogi and Go [Silver et al., 2018], superhuman performance in an entire suite of Atari video games [Mnih et al., 2015,

Hessel et al., 2018, Badia et al., 2020], and impressive robot control [Lin, 1993, Smart and Kaelbling, 2002, Peters et al., 2003, Lillicrap et al., 2015, Higuera et al., 2018, Bellemare et al., 2020], among many other accomplishments.

2.1.1 Value-Based Reinforcement Learning

A common paradigm in RL, known as *value-based* RL, is based on the concept of learning to associate a notion of "value" to each state, and extracting a behavioral policy that should be likely to bring the agent to states with high value. The notion of value in RL is the expected value of the discounted return earned by the agent when following a given policy. Let $(X_i)_{i=0}^{\infty}$ denote the sequence of states visited by an agent in an MDP $(\mathcal{X}, \mathcal{A}, r, P, \gamma)$ starting at state $X_0 = x$ – that is, $X_{k+1} \sim P^{\pi}(\cdot \mid x)$. Given a measure space $(\mathbf{R}_+, \Sigma, \mu)$, the discounted return from state x observed by this trajectory, $G_{\mu}^{\pi}(x)$, is given by

$$G_{\mu}^{\pi}(x) = \int_0^{\infty} \gamma^t r(X_t) d\mu(t) \;\Big|\; X_0 = x \tag{2.2}$$

In discrete time, μ is generally chosen to be the *counting measure* $\#(A) = |A|, A \in \Sigma$, in which case we have

$$G_{\#}^{\pi}(x) = \sum_{i=0}^{\infty} \gamma^i r(X_i) \;\Big|\; X_0 = x \tag{2.3}$$

When μ is the counting measure or the Lebesgue measure, we write $G_{\mu}^{\pi} \triangleq G^{\pi}$, and the context of the problem should immediately resolve the ambiguity. For the remainder of this chapter, we'll consider only the discrete-time setting ($\mu = \#$) for a simpler illustration of the core concepts of RL.

The *value function* $V^{\pi} : \mathcal{X} \to \mathbf{R}$ for an agent following policy $\pi : \mathcal{X} \to \mathscr{P}(\mathcal{A})$[2] is defined as the mapping from states to the expected discounted return:

$$V^{\pi}(x) = \mathop{\mathbf{E}}_{X_i \sim P, \pi} \left[\int_0^{\infty} \gamma^t r(X_t) d\mu(t) \;\Big|\; X_0 = x \right] = \mathop{\mathbf{E}}_{X_{i+1} \sim P^{\pi}(\cdot \mid X_i)} \left[G^{\pi}(X_0) \mid X_0 = x \right] \tag{2.4}$$

[2]The policy should be interpreted as a conditional probability over actions, e.g. $\pi(a \mid x) = \Pr(A_i = a \mid X_i = x)$ for finite state and action spaces.

An observation to note is that (2.4) can be written recursively, like so:

$$\begin{aligned} V^\pi(x) &= \mathop{\mathbf{E}}_{X_i \sim P^\pi}\left[\int_0^t \gamma^s r(X_s) + \gamma^t \int_0^\infty \gamma^s r(X_{s+t}) d\mu(s) \;\middle|\; X_0 = x\right] \\ &= \mathop{\mathbf{E}}_{X_s \sim P^\pi}\left[\int \gamma^s r(X_s) d\mu(s) + \gamma^t V^\pi(X_t) \;\middle|\; X_0 = x\right] \end{aligned} \tag{2.5}$$

In the discrete-time setting we have $\mu = \#$ as discussed above, so we can equivalently write

$$V^\pi(x) = r(x) + \gamma \mathop{\mathbf{E}}_{X' \sim P^\pi}\left[V^\pi(X') \mid X_0 = x\right]$$

The recursive formulation (2.5) is referred to as *the Bellman equation* [Bellman, 1954]. When the state space is finite, in discrete time this can simply be expressed as

$$\boldsymbol{V}^\pi = \boldsymbol{\mathcal{R}} + \gamma \boldsymbol{P}^\pi \boldsymbol{V}^\pi \tag{2.6}$$

where $\mathcal{X} = \{x_i : i = 1, \ldots, |\mathcal{X}|\}$, $\boldsymbol{V}^\pi \in \mathbf{R}^{|\mathcal{X}|}$ given by $\boldsymbol{V}_i^\pi = V^\pi(x_i)$, $\boldsymbol{\mathcal{R}} \in \mathbf{R}^{|\mathcal{X}|}$ satisfies $\boldsymbol{\mathcal{R}}_i = r(x_i)$, and $\boldsymbol{P}^\pi \in \mathbf{R}^{|\mathcal{X}| \times |\mathcal{X}|}$ satisfies $\boldsymbol{P}_{ij}^\pi = P^\pi(x_j \mid x_i)$. It is clear that (2.6) is linear in the value function, and given that $\boldsymbol{P}^\pi$ is a stochastic matrix by construction and $|\gamma| < 1$ by definition, we can compute the value function in closed form:

$$\boldsymbol{V}^\pi = (I - \gamma \boldsymbol{P}^\pi)^{-1} \boldsymbol{\mathcal{R}} \tag{2.7}$$

Note that the inverse $(I - \gamma \boldsymbol{P}^\pi)$ exists since $\boldsymbol{P}^\pi$ is a stochastic matrix and $\gamma \in (0, 1)$ [Puterman, 2014]. Of course, it should be noted that matrix inversion is an expensive operation, rendering the computation of (2.7) intractable when the state space is large.[3] The process of simply determining the value function corresponding to a fixed policy is a computational challenge at the core of value-based RL.

Even if we could compute the value function for a fixed policy, in RL the goal is to find an *optimal* policy. In value-based RL, policies are compared according to their corresponding value functions at each state. It is well-known that the existence of an *optimal* policy according to this criterion is guaranteed [Puterman, 2014] in the discounted infinite horizon

[3] Note that we're often interested in *continuous* state spaces, which have *uncountable* cardinality!

setting[4] and a policy $\pi^\star$ is optimal if

$$\forall\pi\forall x \in \mathcal{X} : V^{\pi^\star}(x) \geq V^\pi(x)$$

Armed with just the knowledge of the value function, determining a good policy is non-trivial (especially if the reward function and the transition probabilities are not given). It is convenient to consider a slightly different mechanism for computing the value function, which makes use of another mapping known as the *action-value function*. For a given policy π, the action-value function (referred to as the Q-function) $Q^\pi : \mathcal{X} \times \mathcal{A} \to \mathbf{R}$ is defined as

$$Q^\pi(x,a) = \mathop{\mathbf{E}}_{X_i,A_i\sim P,\pi}\left[\sum_{i=0}^{\infty}\gamma^i r(X_i) \;\middle|\; X_0 = x, A_0 = a\right] = r(x) + \mathop{\mathbf{E}}_{X'\sim P(\cdot|x,a)}\left[V^\pi(X')\right] \tag{2.8}$$

Naturally, one can construct the value function from the action-value function,

$$V^\pi(x) = \int_{\mathcal{A}} Q^\pi(x,a)\pi(da \mid x) \tag{2.9}$$

Bellman's principle of optimality, which states that optimal policies are Markovian [Bellman, 1957], allows us to characterize the optimal Q-function $Q^\star$ via the recurrence

$$Q^\star(x,a) = r(x) + \mathop{\mathbf{E}}_{X'\sim P(\cdot|x,a)}\left[\max_{a'\in\mathcal{A}} Q^\star(X',a')\right] \tag{2.10}$$

Given the optimal action-value function and a measure space $(\mathcal{A}, \Sigma_A, \nu)$, it is easy to extract an optimal policy:

$$\pi^\star(a \mid x) = \frac{d}{d\nu}\chi_{\arg\max_{a'\in\mathcal{A}} Q^\star(x,a')} \tag{2.11}$$

where χ_M is the characteristic function of a measurable set M, $\frac{d}{d\nu}$ denotes the Radon-Nikodym derivative operator with respect to the reference measure ν. In this thesis, we will mainly be considering MDPs with finite action spaces, so we'll have $\nu = \#$ and an optimal policy is then given by

[4] This refers to the model where agent interacts with the environment indefinitely.

$$\pi^\star(a \mid x) = \frac{1}{|Q^\star(x)|}\mathbf{1}_{\left[a \in \arg\max_{a' \in \mathcal{A}} Q^\star(x,a')\right]}$$
$$|Q^\star(x)| \triangleq \left|\arg\max_{a' \in \mathcal{A}} Q^\star(x,a')\right|$$

2.1.2 Methods for Estimating the Value Function

By our discussion in the previous section, we see that the optimal control problem is easily solved given the knowledge of the optimal (action-) value function. Unfortunately, usually the optimal value function is unknown, and determining the optimal value function can be very challenging. As previously discussed, this can be accomplished by computing a matrix inversion (2.7), however there are several reasons why this is usually not acceptable:

1. Matrix inversion has cubic time complexity, which is too expensive for large or continuous state spaces;
2. This method assumes the agent has knowledge of the precise state transition model and the reward function, which is usually not assumed to be the case.

Next we will look at some alternative methods for estimating the value function, each of which circumvent the expensive matrix inversion.

Policy and Value Iteration

When the state and action spaces are finite and the transition probabilities and reward function are known, the optimal value function and an optimal policy can be approximated efficiently. The *value iteration* algorithm proposed by Bellman [1954] uses dynamic programming [Bellman, 1954] to estimate the optimal value function, and extracts an optimal policy from the estimated optimal value function. More explicitly, if $\boldsymbol{V}^k \in \mathbf{R}^{|\mathcal{X}|}$ represents the estimate of the optimal value function after k iterations, we compute $\boldsymbol{V}^{k+1}$ by an application of the *Bellman optimality operator*,

$$\boldsymbol{V}_x^{k+1} \leftarrow \boldsymbol{R}_x + \gamma \max_{a \in \mathcal{A}} \langle \boldsymbol{P}_{x,a}, \boldsymbol{V}_x^k \rangle \qquad \forall x \in \mathcal{X}$$

where $\boldsymbol{R} \in \mathbf{R}^{|\mathcal{X}|}$ is the vector of rewards at each state and $\boldsymbol{P} \in \mathbf{R}^{|\mathcal{X}|\times|\mathcal{X}|\times|\mathcal{A}|}$ is the matrix

of transition probabilities, where $(\boldsymbol{P}_{x,a})_{x'} = P(x' \mid x, a)$. Upon convergence of the value function, the optimal policy is simply computed by deterministically mapping each state to the action yielding the greatest value according to the value function estimate. It can be shown by a method similar to that shown in §2.1.3 that $\boldsymbol{V}^k$ does indeed converge in $\ell^\infty(\mathbf{R})$ to the optimal value as $k \to \infty$, and for any given error tolerance the number of iterates required grows logarithmically. Each iteration can be computed in $O(|\mathcal{X}|^2|\mathcal{A}|)$ time, making value iteration a substantially more efficient alternative to the matrix inversion method as the state and action spaces grow.

Another simple method to learn the action-value function in the discrete state and action space setting is by *policy iteration* [Howard, 1960]. Unlike value iteration, this method iteratively optimizes the estimate of the optimal policy until convergence while maintaining an estimate of the optimal value function. It begins with a random guess of both the policy and the value function and oscillates between update stages known as *policy evaluation* and *policy improvement*. In the policy evaluation stage, the value function is updated while fixing the current estimate of the optimal policy, and in the policy improvement stage the estimate of the optimal policy is updated while fixing the estimate of the optimal value function. This scheme is conveyed by

$$\boldsymbol{V}_x^{k+1} \leftarrow \boldsymbol{R}_x + \langle \boldsymbol{P}_{x,\boldsymbol{\pi}_x^k}, \boldsymbol{V}_x^k \rangle \qquad \forall x \in \mathcal{X} \quad \text{(Policy Evaluation)} \tag{2.12}$$

$$\boldsymbol{\pi}_x^{k+1} \leftarrow \arg\max_{a \in \mathcal{A}} \left(\boldsymbol{R}_x + \gamma \langle \boldsymbol{P}_{x,a}, \boldsymbol{V}_x^{k+1} \rangle \right) \qquad \forall x \in \mathcal{X} \quad \text{(Policy Improvement)} \tag{2.13}$$

At each iteration, for each state, we compute a value function update requiring $O(|\mathcal{X}|^2)$ (assuming the mapping $x :\mapsto \boldsymbol{\pi}_x^k$ is a constant-time operation), plus a policy update requiring $O(|\mathcal{X}|^2|\mathcal{A}|)$ time, for a total cost of $O(|\mathcal{X}|^2(1 + |\mathcal{A}|))$ per iteration. It is known that the policy iterates of this scheme will converge [Puterman and Brumelle, 1979]. Usually, the state space is much larger than the action space, so iterations of the algorithm are considerably less costly than matrix inversion.

Monte Carlo

When the transition probabilities and the reward function are not known, not all hope is lost. Given a generative model of the environment, we can perform policy evaluation by sampling returns from the generative model and estimating the expected discounted return using an unbiased statistical estimator, such as the sample mean. Suppose the agent

starts at a random state X_0 sampled from a distribution P_0. We sample N trajectories,

$$\begin{aligned} A_i^{(k)} &\sim \pi(\cdot \mid X_i^{(k)}) \\ R_i^{(k)} &= \mathcal{R}(X_i^{(k)}) \\ S_{i+1}^{(k)} &\sim P(\cdot \mid X_i^{(k)}, A_i^{(k)}) \\ G_i^{(k)} &= \sum_{j=i}^{\infty} \gamma^{j-i} R_i^{(k)} \end{aligned}$$

for $k \in \{1, \ldots, N\}$. We then estimate

$$V^{\pi}(X_0) = \frac{1}{N} \sum_{k=1}^{N} G_0^{(k)}$$

This technique is an example of Monte Carlo sampling. While each value estimate is not particularly expensive, Monte Carlo methods are known to exhibit high variance [Sutton and Barto, 2018]. Consequently, many samples from the environment are generally required to attain high-fidelity value estimates. In particular, in order to estimate the action-value function to within an pointwise error of at most $\epsilon > 0$ with probability $1 - \delta$ for $\delta \in (0, 1)$, we must have

$$N \geq \frac{c}{(1-\gamma)^3} \frac{|\mathcal{X}||\mathcal{A}| \log(c|\mathcal{X}||\mathcal{A}|/\delta)}{\epsilon^2}$$

where c is a universal constant and N is the required number of samples starting from each state [Agarwal et al., 2019].

Temporal Difference Learning

In order to circumvent the high variance of Monte Carlo methods, when the transition probabilities are not known an alternative method for estimating the value function is *temporal difference learning* [Sutton, 1988]. In this setting, we begin with an arbitrary initialization of the value function V and perform updates after every state transition. That is, at state x we sample

$$
\begin{aligned}
a &\sim \pi(\cdot \mid x) \\
x' &\sim P(\cdot \mid x, a) \\
r &= \mathcal{R}(x)
\end{aligned}
$$

and then we update V according to

$$V(x) = \alpha(r + \gamma V(x')) + (1 - \alpha)V(x)$$

for some learning rate $\alpha \in (0, 1)$. The trick here is that we use our current "guess" of the value function as our estimate of the tail of the expected return (we are correcting our guess against a target generated by our guess). This is called *bootstrapping*, and estimating the expected return with bootstrapping incurs a bias. Consequently, in accordance with the bias-variance tradeoff, the variance of temporal difference learning algorithms tends to be far lower than in Monte Carlo algorithms, which theoretically suggests that less samples are needed for an accurate (biased) estimate of the value function. Moreover, it is known that this bias diminishes and ultimately vanishes as the agent observes more and more data from the environment [Sutton and Barto, 2018], which further demonstrates that temporal difference learning is a viable technique.

2.1.3 Contraction Arguments

A recurring motif in the reinforcement learning literature is the use of *contraction arguments* to prove that a sequence of value function iterates converges to a fixed point. Generally, this involves defining an operator that is *contractive* on the space of value functions for instance, and ultimately invoking the *Banach fixed point theorem*. A brief example will be given. First, we define what it means for an operator to be contractive. Refer to Appendix A, particularly §A.1, for a primer on the topological concepts used in this section.

Definition 2 (Contraction mapping)**.** Let $\mathcal{T} : X \to X$ be an operator on a metric space (X, d). We say that $\mathcal{T}$ is a *contraction mapping* ($\mathcal{T}$ is *contractive*) if for any pair of points $(x, y) \in X \times X$ we have

$$d(\mathcal{T}x, \mathcal{T}y) \leq \gamma d(x, y)$$

where $\gamma \in [0, 1)$. ▽

Now we're ready to state the celebrated fixed point theorem of Banach.

Theorem 2.1 (Banach's fixed point theorem)**.** *Let (X, d) be a complete metric space, and let $\mathcal{T} : X \times X$ be contractive. Then the sequence $\{x_k\}_{k=1}^{\infty}$ where $x_k = \mathcal{T}^k(x_1)$ converges to a fixed point $x \in X$, in the sense that $\mathcal{T}x = x$. Moreover, this fixed point is unique.*

Proof. For any $x \in X$, we have

$$\begin{aligned} d(x_n, x_m) &\leq d(x_n, \mathcal{T}^n x) + d(x_m, \mathcal{T}^n x) \\ &\leq \gamma^n d(x_1, x) + \gamma^n d(x_{m-n}, x) \\ &\to 0 \end{aligned}$$

Therefore $\{x_k\}_{k=1}^{\infty}$ is a Cauchy sequence. Since (X, d) is complete, it follows that $\{x_k\}_{k=1}^{\infty}$ converges to some $x^\star \in X$. Since $[0, \infty)$ is known to be complete [Kreyszig, 1978], the sequence $\{d_k\}_{k=1}^{\infty}$ where $d_k = d(x_k, x^\star)$ converges to 0, so we see that $d(\mathcal{T}x^\star, x^\star) = 0$. By the separation of points property (definition 23), we must have $\mathcal{T}x^\star = x^\star$. The uniqueness of the fixed point follows from Lemma A.1. □

We conclude the section by demonstrating a method for learning the value function using applications of the Bellman equation.

Theorem 2.2 (Sutton and Barto [2018])**.** *Consider an MDP $(\mathcal{X}, \mathcal{A}, r, P, \gamma)$ and a policy $\pi : \mathcal{X} \to \mathcal{P}_p(A)$, where $\gamma < 1$ and r is bounded. Denote by $\mathcal{V}$ the set of all value functions $V : \mathcal{X} \to \mathcal{R}$ where $\mathcal{R} = [\frac{\min_x r(x)}{1-\gamma}, \frac{\max_x r(x)}{1-\gamma}]$. We define the Bellman operator $\mathcal{T}^\pi : \mathcal{V} \to \mathcal{V}$ according to*

$$(\mathcal{T}^\pi V)(x) = \int_{\mathcal{X} \times \mathcal{A}} (r(x) + \gamma V(x'))\, dP(x' \mid x, a) d\pi(a)$$

There exists a fixed point V^π for $\mathcal{T}^\pi$, and the sequence $\{V_k\}_{k=1}^{\infty}$ converges to V^π uniformly.

Proof. Endow $\mathcal{V}$ with the metric d given by $d(V_1, V_2) = \sup_{x \in \mathcal{X}} |V_1(x) - V_2(x)|$. We will begin by showing that $\mathcal{T}^\pi$ is contractive. We have

$$
\begin{aligned}
d(\mathcal{T}^\pi V_1, \mathcal{T}^\pi V_2) &= \sup_{x\in\mathcal{X}} \int_{\mathcal{X}\times\mathcal{A}} |r + \gamma V_1(x') - (r(x) + \gamma V_2(x'))| dP(x' \mid x, a) d\pi(a) \\
&= \sup_{x\in\mathcal{X}} \int_{\mathcal{X}\times\mathcal{A}} \gamma |V_1(x') - V_2(x')| dP(x' \mid x, a) d\pi(a) \\
&\leq \gamma \sup_{x\in\mathcal{X}} |V_1(x) - V_2(x)| \int_{\mathcal{X}\times\mathcal{A}} dP(x' \mid x, a) d\pi(a) \\
&= \gamma \sup_{x\in\mathcal{X}} |V_1(x) - V_2(x)| \\
&= \gamma d(V_1, V_2)
\end{aligned}
$$

Since $\gamma \in [0, 1)$, $\mathcal{T}^\pi$ is indeed a contraction map. Moreover, it is known that the metric space $(\mathcal{V}, d)$ presented here is complete when the image of $\mathcal{V}$ is complete, which is the case since $\mathcal{R}$ is compact [Kreyszig, 1978]. Therefore, by the Banach fixed-point theorem, the sequence $\{V_k\}_{k=1}^\infty$ where $V_k = (\mathcal{T}^\pi)^k V_1$ converges to a value function V^π which satisfies $\mathcal{T}^\pi V^\pi = V^\pi$, and V^π is the unique solution to the fixed point equation for $\mathcal{T}^\pi$. Moreover, the convergence is uniform since we have $|V_n(x) - V^\pi(x)| \leq \gamma^n d(V_1, V^\pi)$ independently of the state x. □

2.1.4 Deep Q Networks

To conclude this brief overview of reinforcement learning, we'll discuss a particularly successful algorithm that makes the basis of many state of the art value-based learning algorithms that are around today. This algorithm, known as *Deep Q-learning*, was famously employed by the DQN architecture to train an RL agent to outperform humans in Atari video games [Mnih et al., 2015].

The idea is founded on the concept of Q-learning, which is an RL algorithm that was classically studied with MDPs having finite state and action spaces [Watkins, 1989]. Rather than learning the value function, in Q-learning we learn a related concept called the *action-value function* $Q : \mathcal{X} \times \mathcal{A} \to \mathcal{R}$ for an MDP $(\mathcal{X}, \mathcal{A}, r, P, \gamma)$ with $\mathsf{Ran}(r) \subset \mathcal{R}$. This function is defined as

$$
Q(x, a) = \mathbf{E}\left[\sum_{k=0}^{\infty} \gamma^k r(x_k) \,\middle|\, x_0 = x, a_0 = a \right] = r(x) + \gamma \mathop{\mathbf{E}}_{x'\sim P(\cdot|x,a)} [V(x')] \tag{2.14}
$$

The Q-learning algorithm learns the action-value function (also known as the Q-function)

by approximate dynamic programming by iteratively applying the *Bellman optimality operator* $\mathscr{T}^\star$,

$$\mathscr{T}^\star Q(x,a) = r(x) + \gamma \mathop{\mathbf{E}}_{x' \sim P(\cdot|x,a)} \left[\max_{a' \in \mathcal{A}} Q(x', a') \right] \tag{2.15}$$

Denardo [1967] shows that $\mathscr{T}^\star$ is contractive, and so it has a unique fixed point often called $Q^\star$. However, in many applications of interest, the transition matrix P is unknown, so (2.15) cannot be computed. Fortunately, iterates of (2.15) estimated by data sampled from the environment are also shown to converge, resulting in Algorithm 1.

Algorithm 1 Q-Learning

Require: Q^i, the Q-function estimated after the ith iteration
Require: $\alpha_i \in \mathbf{R}_+$, the "learning rate" or exponential smoothing parameter
Require: (s, a), a state-action pair.
$Q^{i+1}(x,a) \leftarrow Q^i(x,a) \quad \forall (x,a)$
$x', r \sim P(\cdot, \cdot \mid x, a)$ ▷ Sample next state from the environment
$a' \sim \mathsf{U}\left(\arg\max_{a' \in \mathcal{A}} Q(x', a')\right)$ ▷ Pick next best action
$Q^{i+1}(x,a) \leftarrow \alpha_i (r + \gamma Q^i(x', a') - Q^i(x,a))$
return Q^{i+1}

Under certain conditions on the sequence $\{\alpha_i\}_{i=1}^\infty$, it is known that $\{Q_i\}_{i=1}^\infty \to Q^\star$ [Bertsekas and Tsitsiklis, 1996] when $\{Q_i\}_{i=1}^\infty$ is constructed by applications of Algorithm 1.

An important observation is that the updates in Algorithm 1 are independent of the policy that governs the behavior of the agent in the environment. Consequently, we can perform the Q-Learning update with *any* transition (x, a, x') sampled from P. Such algorithms are said to be *off-policy*, and are beneficial in the sense that we may collect a dataset of transitions and perform updates using this dataset as many times as we'd like, which makes better use of the data than simply applying an update using an observed transition and then forgetting about that transition.

Deep Q-learning is a framework for performing Q-learning updates in environments with large (even infinite) state spaces, by means of approximating Q functions by deep neural networks. The DQN implementation takes advantage of the off-policy nature of Q-learning, and incorporates some additional techniques to stabilize the learning process.

Firstly, DQN makes use of *experience replay* [Lin, 1992], which consists of storing observed state transitions in a buffer and performing Q-learning updates on minibatches sampled from the buffer. As well as improving data efficiency as discussed above, experience re-

play is also particularly helpful when training neural networks with stochastic samples. In order to learn an expected value from samples (such as the Q-function), convergence guarantees can only be made when the samples used for training are independent and identically-distributed (i.i.d.) [Friedman, 2017]. This is generally not the case in RL, because consecutive trajectory samples are highly dependent on each other. By accruing a buffer of transitions and sampling from the buffer uniformly when performing neural network updates, transitions are far less dependent on each other. If we keep the policy fixed, then in the limit as the buffer contains ininitely-many transitions, all samples from the buffer are distributed according to the stationary distribution of the Markov chain induced by the policy. Therefore, experience replay helps by providing the deep neural networks with approximately i.i.d. data.

Another improvement included in DQN is the use of a *target network*. Since the Q-function estimate is constantly evolving over the course of training, the supervised learning problem of mapping $Q \mapsto r+\gamma Q$ is non-stationary. This severely hinders our ability to perform policy evaluation. Another tactic to minimize the non-stationarity involves additionally maintaining a "target Q-network" that is updated at a much slower rate. The target network is used to compute the regression targets, so it is approximately fixed from the perspective of the predictive Q-network used for inducing a policy. A sketch of DQN is given in Algorithm 2.

Algorithm 2 DQN, Mnih et al. [2015]

Require: Q^1_θ, Q^1_ϕ, neural nets parameterized by θ^1, ϕ^1
Require: $\{\alpha_i\}_{i=1}^{\infty}$, sequence of learning rates
Require: $\tau \in (0, 1)$, exponential smoothing parameter
Require: π, a policy mapping Q-values to a probability distribution over actions
  $B \leftarrow \emptyset$ ▷ Initialize replay buffer
  $x_1 \sim \rho$ ▷ Sample start state from environment
  for $k \in \mathbf{N}$ **do**
    $a_k \sim \pi(Q^k_\theta(x, \cdot))$
    $(x_{k+1}, r_k) \sim P(\cdot, \cdot \mid x_k, a_k)$
    $B \leftarrow B \cup \{(x_k, a_k, r_k, x_{k+1})\}$ ▷ Add transition sample to replay buffer
    $\theta^{k+1} \leftarrow \theta^k - \alpha_k \nabla_\theta \left(r_k + \gamma \arg\max_{a \in \mathcal{A}} Q^k_\phi(x', a) - Q^k_\theta(x_k, a_k)\right)^2$
    $\phi^{k+1} \leftarrow \tau\phi^k + (1-\tau)\theta^{k+1}$
    if episode is over **then**
      $x_{k+1} \sim \rho$ ▷ Reset the environment
    end if
  end for

2.2 Stochastic Processes and Differential Equations

The material in this section makes extensive use of terminology and results from measure theory and stochastic process theory. An overview of these concepts is given in Appendices B and C.

Stochastic processes are simply trajectories of random variables through time. They will be essential in several developments later in the thesis, and will form the basis for the analysis of the return distributions.

In order to formalize this concept, we have to introduce the notion of a *filtration*. A filtration provides us with a mechanism of "evolving a probability space over time", so that the event space can in some sense reflect the history of the data previously observed.

Definition 3 (Filtration, Le Gall [2016])**.** Let $(\Omega, \mathcal{F}, \Pr)$ be a probability space. A *filtration* of $\mathcal{F}$ is a collection $(\mathcal{F}_t)_{t\geq 0}$ of σ-algebras where $\mathcal{F}_t \subset \mathcal{F}$ for each t, and $\mathcal{F}_s \subset \mathcal{F}_t$ whenever $s < t$. A probability space associated with a filtration is called a *filtered probability space*, and is written as the 4-tuple $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \Pr)$. $\triangledown$

A Stochastic Differential Equation (SDE) is similar in principle to a differential equation seen in a standard calculus course, however an SDE involves stochastic processes. Consider a random process $(X_t)_{t\geq 0}$ on a filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \mu)$, where $X_t : \mathcal{F}_t \to \mathcal{X}$ is a random variable (it is $\mathcal{F}_t$-measurable) and $\mathcal{X}$ is an arbitrary space that X_t takes values in. For a function $f : \mathcal{X} \to \mathcal{X}$, we can study the following SDE,

$$Y_t = \int_0^t f(Y_s) dX_s, \tag{2.16}$$

where the integral is taken with respect to the stochastic process $(X_t)_{t\geq 0}$, and for this thesis it is understood as the Itô stochastic integral.

Definition 4 (Itô Integral, Le Gall [2016])**.** Consider a filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \Pr)$. Let $(X_t)_{t\geq 0} \subset \Omega$ be a continuous semimartingale (Definition 40), and let $f : \Omega \to \Omega$ be defined such that the mappings $t \mapsto f(X_t(\omega))$ are continuous for each $\omega \in \mathcal{F}_t$. The *Itô integral* $\int_0^t f(X_s) dX_s$ is given by

$$\int_0^t f(X_s) dX_s = \lim_{n\to\infty} \sum_{i=1}^{p_n - 1} f(X_{t_i^n})(X_{t_{i+1}^n} - X_{t_i^n})$$

where $\{t_i^n\}_{i=1}^{p_n-1}$ is a partition of $[0,t]$, and the sequence of partitions indexed by n have mesh tending to 0 as $n \to \infty$. $\triangledown$

Note that $(Y_t)_{t\geq 0}$ defined in (2.16) is itself a stochastic process. It is common to express SDEs in a "differential form"; for instance (2.16) may be written as

$$dY_t = f(Y_t)dX_t$$

although such expressions are merely symbolic.

The remainder of this section will give a general overview of the concepts from the theory of stochastic processes and differential equations that are helpful for understanding the remainder of the thesis. Note that the concepts in this section will not be presented with full mathematical rigor, as that would likely require a book on its own [Le Gall, 2016].

2.2.1 Brownian Motion

Brownian motion is ubiquitous in the study of stochastic processes. The idea can be motivated as follows.

Let $X_0 \triangleq 0 \in \mathbf{R}$. Suppose we are modeling the trajectory of the random process $(X_t)_{t\geq 0}$, where X is "continuously perturbed" by Gaussian noise with mean 0. What does it mean for something to be *continuously perturbed* by noise? A natural way to reason about this is to discretize time, and suppose that the variable at consecutive timesteps differs by a random quantity sampled independently from a Gaussian with zero mean. We want X_1 to have variance 1, and we want this variance to spread evenly through time in the sense that X_t has variance t. We can begin with a very coarse discretization where the timestep τ has duration 1, which involves sampling $X_1 \sim \mathcal{N}(0,1)$ and interpolate linearly form $t=0$ to $t=1$. Then we can study the behavior as $\tau \to 0$. For any $\tau > 0$, we simply sample $X_{t+\tau} \sim X_t + \mathcal{N}(0,\tau)$. Alternatively, we can sample $(X_{k\tau})_{k\in\mathbf{N}}$ via a Gaussian process with covariance kernel $K(X_s, X_t) = \min(s,t)$ [Williams and Rasmussen, 2006]. Figure 2.1 illustrates some of these samples for various values of τ.

Considering once again the filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \mu)$, the criteria for a Brownian motion $(B_t)_{t\geq 0}$ can be stated formally as

1. $B_0 = 0$, μ-almost surely;

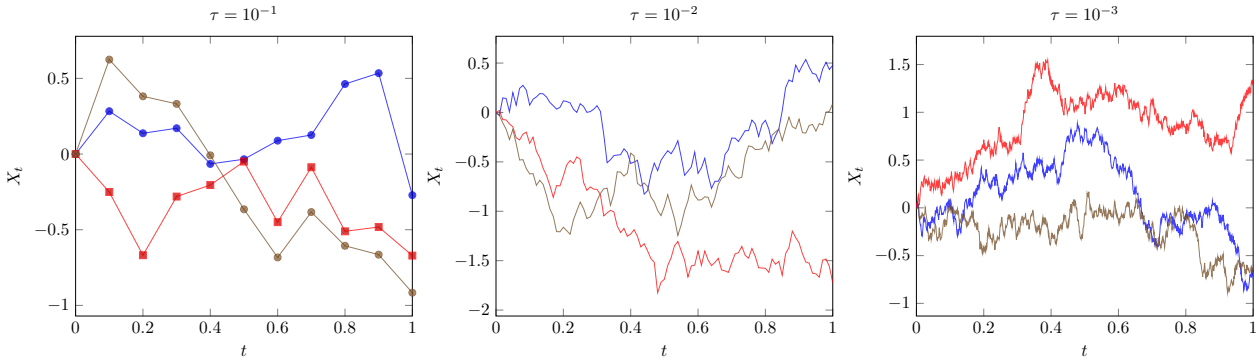


Figure 2.1: Discretized Brownian motion trajectories for various timesteps τ

2. For any $0 \leq r < s < t$, the random variable $B_t - B_s$ is independent from $\mathcal{F}_r$ and is distributed according to $\mathcal{N}(0, t-s)$;

3. The *sample paths* of $(B_t)_{t\geq 0}$, defined as the mappings $t \mapsto B_t(\omega)$ for any fixed $\omega \in \mathcal{F}_t$, are continuous.

Proving that such a process exists is not trivial by any means. Fortunately, Brownian motion *does* exist, and Le Gall [2016] can be consulted for its construction.

2.2.2 The Expressivity of Itô Diffusions

A recurring motif in this thesis will consist of a special type of stochastic differential equation known as an *Itô diffusion*. These are the SDEs of the following form,

$$dX_t = a(t, X_t)dt + b(t, X_t)dB_t. \tag{2.17}$$

Since the only source of noise in these types of processes is Gaussian, it may appear at first glance that the class of solutions to Itô Diffusion SDEs is fairly limited. However, it turns out that processes of this form converge to a very rich class of stationary distributions. This is nicely stated in the celebrated *Martingale Representation Theorem*.

Theorem 2.3 (Martingale Representation Theorem, Le Gall [2016]). *Consider a filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \mu)$, where $(\mathcal{F}_t)_{t\geq 0}$ is the completed[5] canonical filtration of a Brownian motion $(B_t)_{t\geq 0}$ where $B_0 = 0$ almost surely. For any random variable $Z \in L^2(\Omega, \mathcal{F}_\infty, \mu)$, there exists a unique square-integrable progressive process $(h_t)_{t\geq 0}$ such that*

[5]The *completion* of a σ-algebra $\mathcal{F}$ is the σ-algebra generated by $\mathcal{F}$ together with all subsets of sets $A \in \mathcal{F}$ that have measure 0 [Le Gall, 2016].

$$Z = \mathbf{E}[Z] + \int_0^\infty h_t dB_t. \tag{2.18}$$

Recall the definition of the random return. Measuring time with respect to the Lebesgue measure, the return is given by

$$G^\pi(x) = \int_0^T \gamma^t r(X_t) dt.$$

Since r is bounded and $\gamma < 1$, $G^\pi(x)$ is obviously square integrable. Moreover, Brownian motion has infinite variation, and particles under Brownian motion spread evenly over $\mathbf{R}$. Therefore, as long as Law $(G^\pi(x))$ is absolutely continuous with respect to the Lebesgue measure, we'll have

$$G^\pi(x) \in L^2(\Omega, \mathcal{F}_\infty, \Pr)$$

where $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \Pr)$ is the probability space P defined above. Therefore, *any* return distribution that is absolutely continuous with respect to the Lebesgue measure can be expressed as the stationary distribution of an Itô diffusion.

The most difficult part when dealing with stochastic differential equations (SDEs) is, unsurprisingly, the stochastic integral. It would be desirable then if we could express a random variable as the solution to an SDE of the form

$$dZ_t = -\nabla\phi(t, Z_t)dt + dB_t \tag{2.19}$$

for some twice differentiable function ϕ. The SDE (2.19) is an example of *Langevin dynamics*, and the method of representing random variables via Langevin dynamics has become popular in the machine learning literature in recent years [Welling and Teh, 2011, Wibisono, 2018, Raginsky et al., 2017]. In the context of reinforcement learning, Zhang et al. [2018] represents the evolution of a parameterized policy as a set of particles in parameter space under the influence of Langevin dynamics. Moreover, Martin et al. [2020] exhibits a similar technique in the discrete-time value-based setting.

Let $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0})$ be a filtered measurable space on which $(B_t)_{t\geq 0}$ is a Brownian motion

with respect to a probability measure μ. The following theorem will enable us to transform a Brownian motion to Langevin dynamics.

Theorem 2.4 (Girsanov's Theorem, [Girsanov, 1960])**.** *Let $(\Omega, \mathcal{F})$ be a measurable space and let $(\mathcal{F}_t)_{t\geq 0}$ be a filtration of $\mathcal{F}$. Suppose μ, ν are measures on $(\Omega, \mathcal{F})$ which are absolutely continuous with respect to one another on $\mathcal{F}_\infty$. Let $(D_t)_{t\geq 0}$ be the martingale with càdlàg[6] sample paths such that for every $t \geq 0$ we have*

$$D_t = \frac{d\nu}{d\mu|_{\mathcal{F}_t}}$$

where $\mu|_{\mathcal{F}_t} : \mathcal{F}_t \to \mathbf{R}_+$ is the measure μ restricted to the domain $\mathcal{F}_t$.

Let $([L, L]_t)_{t\geq 0}$ denote the quadratic variation of $(L_t)_{t\geq 0}$ (shown in Definition 43 of Appendix C). Assume that D has continuous sample paths, and let L be the continuous local martingale such that

$$D_t = \exp\left(L_t - \frac{1}{2}[L, L]_t\right)$$

Then if M is a continuous local martingale under μ, $(M - [M, L]_t)_{t\geq 0}$ is a continuous local martingale under ν, where $([M, L]_t)_{t\geq 0}$ is the bracket of $(M_t)_{t\geq 0}, (L_t)_{t\geq 0}$ as shown in Definition 44 of Appendix C.

Let ν be a probability measure on $(\Omega, \mathcal{F})$ that is absolutely continuous with respect to μ. We'll additionally impose the constraint that $\nu_1 = \text{Law}(Z_1)$. Girsanov's theorem tells us that

$$\frac{d\nu}{d\mu} = \exp\left(-\int_0^1 \langle \nabla\phi(B_t, t), dB_t\rangle - \frac{1}{2}\int_0^1 \|\nabla\phi(B_t, t)\|^2 dt\right) \tag{2.20}$$

Let $\mathcal{M}_{Z_1} = \{\nu \in \mathcal{P}_p(\Omega) : \nu_1 = \text{Law}(Z_1)\}$, where $\mathcal{P}_p(\Omega)$ is the set of probability measures on $(\Omega, \mathcal{F})$. Then we'll define our target measure $\nu^\star$ by

$$\nu^\star = \arg\min_{\nu\in\mathcal{M}_{Z_1}} D_{\text{KL}}(\nu \parallel \mu)$$

[6]This refers to functions that are *continue à droite, limites à gauche* – that is – right continuous functions with left limits.

The reason for this specification is that the chain rule of the KL divergence yields

$$\begin{aligned} D_{\mathrm{KL}}\left(\nu^{\star} \parallel \mu\right) &= \arg\min_{\nu \in \mathcal{M}_{Z_1}} \left[D_{\mathrm{KL}}\left(\nu_1 \parallel \mu_1\right) + \int \nu_1(dx) D_{\mathrm{KL}}\left(\nu(\cdot \mid Z_1 = x) \parallel \mu(\cdot \mid Z_1 = x)\right) \right] \\ &= D_{\mathrm{KL}}\left(\mathrm{Law}(Z_1) \parallel \mu_1\right) \end{aligned}$$

where the final step follows since the only constraint on ν occurs at ν_1, so we can minimize the KL divergence by ensuring that $\mu = \nu$ almost everywhere. But since Brownian motion has independent increments and $\mu = \nu$ almost everywhere, it follows that

$$\frac{d\nu}{d\mu} = \varrho(Z_1)$$

for some function ϱ. Consequently, we see that $\frac{d\nu}{d\mu}$ must correspond to the density of Z_1.

Proceeding, we'd like to find a way to eliminate the stochastic integral from (2.20). Fortunately, we can do so by exploiting Itô's lemma. Let $U_t = \phi(B_t, t)$. Then

$$dU_t = \frac{\partial \phi}{\partial t}(B_t, t)dt + \frac{1}{2}\Delta\phi(B_t, t) + \langle \nabla\phi(B_t, t), dB_t \rangle$$

Note that the final term above is equivalent to the integrand in (2.20). Integrating yields

$$U_1 = U_0 + \int_0^1 \left[\frac{\partial \phi}{\partial t}(B_t, t) + \frac{1}{2}\Delta\phi(B_t, t) \right] dt + \int_0^1 \langle \nabla\phi(B_t, t), dB_t \rangle$$

Upon substitution into (2.20), we have

$$\frac{d\nu}{d\mu} = \exp\left(U_0 - U_1 + \int_0^1 \left[\frac{\partial \phi}{\partial t}(B_t, t) + \frac{1}{2}\Delta\phi(B_t, t) - \frac{1}{2}\|\nabla\phi(B_t, t)\|^2 \right] dt \right) \tag{2.21}$$

Recall that $\frac{d\nu}{d\mu}$ should be expressed as a function of $Z_1 = B_1$ only. Since $B_0 = 0$ almost surely, it follows that the integral in (2.21) must vanish. This presents us with the following characterization of the function ϕ such that $\text{Law}(Z_1) = \nu_1$,

$$\frac{\partial \phi}{\partial t} + \frac{1}{2}\Delta\phi - \frac{1}{2}\|\nabla\phi\|^2 = 0 \tag{2.22}$$

Letting $h \triangleq e^{-\phi}$, we see that

$$\begin{aligned}
0 &= -\frac{\partial \log h}{\partial t} - \frac{1}{2}\Delta(\log h) - \frac{1}{2}\|\nabla(\log h)\|^2 \\
&= \frac{1}{h}\frac{\partial h}{\partial t} + \frac{1}{2}\nabla\cdot\frac{\nabla h}{h} + \frac{1}{2}\left\|\frac{1}{h}\nabla h\right\|^2 \\
&= \frac{1}{h}\frac{\partial h}{\partial t} + \frac{1}{2}\left[\frac{1}{h}\Delta h - \frac{1}{h^2}\|\nabla h\|^2\right] + \frac{1}{2}\left\|\frac{1}{h}\nabla h\right\|^2 \\
&= \frac{\partial h}{\partial t} + \frac{1}{2}\Delta h
\end{aligned}$$

Therefore, $h = e^{-\phi}$ satisfies the *heat equation* [Harrison, 2013, Ullrich, 2011]! Explicitly, we want to solve the following boundary value problem for a function $h : \mathcal{X} \times [0,1] \to \mathbf{R}$,

$$\begin{cases} \frac{\partial h}{\partial t} + \frac{1}{2}\Delta h = 0 \\ h(x,1) = \psi(x) \end{cases} \tag{2.23}$$

where ψ encodes a terminal condition. The Feynman-Kac formula [Kac, 1949] (discussed in §C.3) famously shows[7] that (2.23) is satisfied by

$$h(x,t) = \mathbf{E}\left[\psi(B_1) \mid B_t = x\right] \tag{2.24}$$

And equivalently, this provides us with the following characterization of ϕ,

$$\phi(x,t) = -\log \mathbf{E}\left[\psi(B_1) \mid B_t = x\right] \tag{2.25}$$

When (2.25) is satisfied, (2.21) becomes

[7] In fact, (2.23) is simply a *Kolmogorov backward equation*, which will be discussed in more detail later in the thesis.

$$\begin{aligned}\frac{d\nu}{d\mu} &= \exp(U_0 - U_1) \\ &= \exp(\phi(B_0, 0) - \phi(B_1, 1)) \\ &= \exp(\log \psi(B_1)) \\ &= \psi\end{aligned}$$

This tells us that $\psi \equiv \varrho$, the density of $B_1 = Z_1$. Finally, we see that the stationary solution of the SDE

$$dZ_t = \nabla \log \mathbf{E}\left[\varrho(Z_1) \mid Z_t\right] dt + dB_t \tag{2.26}$$

will have density ϱ.

Remark 2.1. Notably, no convexity assumptions are necessary for this convergence.

Remark 2.2. Unlike sampling algorithms based on Markov Chain Monte Carlo (MCMC) such as Metropolis Hastings [Hastings, 1970], we have a guarantee that at time $t = 1$ we'll have *an exact sample* from ϱ. However, in many cases, computing $\nabla \mathbf{E}\left[\varrho(Z_1) \mid Z_t\right]$ exactly will not be tractable, and additionally simulating (2.26) will accrue error due to time discretization.

In the statistics and machine learning literature, sampling via such Langevin diffusions is common practice [Roberts and Stramer, 2002]. The technique has been used in particular to extend existing algorithms to a Bayesian treatment. The Stochastic Gradient Langevin Diffusion (SGLD) framework [Welling and Teh, 2011], for example, simulates Langevin diffusions in parameter space to produce a Bayesian extension of Stochastic Gradient Descent (SGD), which is used to compute posterior distributions over large parameterized models.

2.3 Continuous-Time Dynamics

So far, we've discussed algorithms that learn control policies that evolve in discrete timesteps. While every computer program must evolve discretely in time, we may be interested (and we will be for the sake of this thesis) in how reinforcement learning algorithms behave as the timestep duration tends to 0. Mathematics become far more technical in this regime,

since the infinitesimal differentials between some quantities may not be well defined. As we'll see below, even in the absense of any uncertainty, continuous-time optimal control presents considerable difficulties.

2.3.1 The Deterministic Case

A well-known result from optimal control theory is that when the optimal value function is differentiable, it satisfies the following equation,

$$V(x)\log\gamma + \sup_{a\in\mathcal{A}}\left[r(x,a) + \langle\nabla V(x), f(x)\rangle\right] = 0 \tag{2.27}$$

where $\mathcal{A}$ is the action space, $r : \mathcal{X}\times\mathcal{A}\to\mathbf{R}$ is the reward function, $\dot{x}(t) = f(x(t), a(t))$, and the optimal value function $V(x)$ is defined as

$$V(x) = \sup_{a(\cdot)\in L^1(\mathbf{R}_+} \int_0^\infty \gamma^t r(x(t), a(t))dt \qquad x(0) = x \tag{2.28}$$

Equation (2.27) is known as the *Hamilton-Jacobi-Bellman* (HJB) equation [Fleming and Soner, 2006].[8] The HJB equation takes on a similar form to the Bellman equation, but we note one immediate difference: the Bellman equation expresses V recursively, and the HJB equation expresses it *differentially*. This is to be expected, however, since the HJB equation can be interpreted as the limiting equation when consecutive states are separated by an infinitesimal time in the Bellman equation. Of course, that only makes sense if we **assume that the universe does not stop and wait for us in between timesteps of an MDP**.

Rather than a recurrence relation, the HJB equation presents us with a nonlinear partial differential equation (PDE). PDEs are notoriously difficult to solve, so already continuous-time RL looks discouraging. However, the challenges do not stop there. Recall that the optimal value function satisfies (2.27) *if it is differentiable*, but what happens when it is not differentiable? This may seem like merely a technicality for at least a couple of reasons:

1. It may seem like we can just avoid the issue altogether by approximating the value function somehow, and possibly weakening the notion of solutions to (2.27);

[8] Some texts refer to this equation as the *dynamic programming equation*. However, "dynamic programming" refers to an algorithmic technique for solving such equations [Bellman, 1954], so we prefer to avoid this nomenclature.

2. Perhaps the consequences of assuming the value function is differentiable in practice are negligible?

We will proceed by showing that this phenomenon is in fact problematic both in theory and in practice.

The following is an example due to Munos [2004]. Consider a continuous-time MDP where

- The state space $\mathcal{X} = [0, 1]$;
- The action space $\mathcal{A} = \{-1, 1\}$;
- The dynamics are given by $\dot{x}(t) = a(t)$ where $x(\cdot)$ is the state signal and $a(\cdot)$ is the control signal;
- The reward function is $r(x) = \delta_0(x) + 2\delta_1(x)$;
- Episodes run until the agent exits the interior of $\mathcal{X}$.

This problem seems exceptionally simple at first glance. So simple, in fact, that we can easily plot the optimal value function, as shown in Figure 2.2.

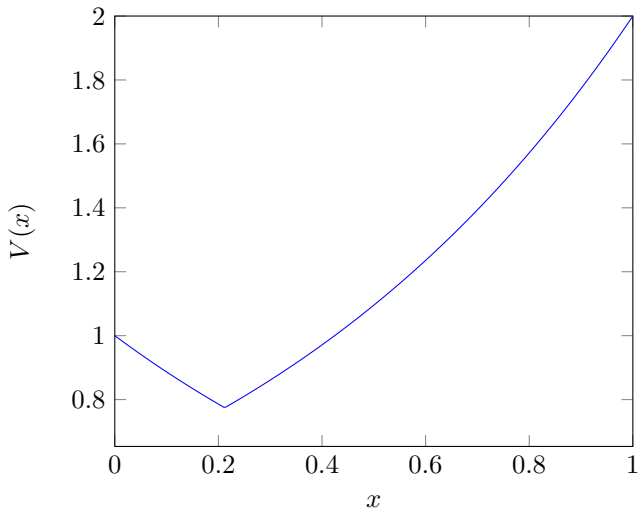


Figure 2.2: Value function from Munos' toy problem, $\gamma = 0.3$

As Figure 2.2 indicates, the value function for this problem is not differentiable everywhere. We'll proceed by assessing our first skepticism: why can't we just approximate the value function by a differentiable function? As a matter of fact, we *can* do this, however the approximation is far from trivial. Looking at 2.2, it is reasonable to suggest that we simply search for V among piecewise differentiable functions, or even Sobolev spaces.

Clearly in this case we'd still find the value function, as it is piecewise differentiable. However, this absolutely does not solve the problem, not even for this one example environment: it is well-known that there are infinitely many functions differentiable almost everywhere that satisfy (2.27) [Fleming and Soner, 2006]! Munos [2004] demonstrates that solutions to the HJB equation can be vastly dissimilar to the value function. Consequently, attempting to search for a solution to the HJB equation among functions that are differentiable almost everywhere using gradient based methods is likely to converge to a local optimum, and the local optimum can be entirely unrepresentative of the true value function.

Fortunately, not all hope is lost. The celebrated work of Crandall and Lions [1983] introduces a weak interpretation of solutions to the HJB equation, namely *viscosity solutions*, for which there exists a unique solution among the space of almost-everywhere-differentiable functions.

Definition 5 (Viscosity solution, [Crandall and Lions, 1983])**.** Let $\mathcal{O} \subset \mathbf{R}^d$ be an open set. We define set-valued functions $E_+, E_- : C(\mathcal{O}) \to 2^{\mathcal{O}}$ according to

$$E_+(\psi) = \{y \in \mathcal{O} : y \in \arg\max \psi \wedge \psi(y) > 0\}$$
$$E_-(\psi) = \{y \in \mathcal{O} : y \in \arg\min \psi \wedge \psi(y) < 0\}$$

We consider equations of the form $H(y, V, \nabla V) = 0$ for a continuous function $H : \mathcal{O} \times \mathbf{R} \times \mathbf{R}^d \to \mathbf{R}$. A function $v \in C(\mathcal{O})$ is said to be a **viscosity subsolution** of $H(y, V, \nabla V) = 0$ if for every positive function $\phi \in C_c^\infty(\mathcal{O})$ and $k \in \mathbf{R}$ we have

$$E_+(\phi(v-k)) \neq \emptyset \implies \exists y \in E_+(\phi(v-k)) : \\ H\left(y, v(y), -\frac{v(y)-k}{\phi(y)}\nabla\phi(y)\right) \leq 0$$

Similarly, a function $v \in C(\mathcal{O})$ is called a **viscosity supersolution** of $H(y, v, \nabla V) = 0$ if for every positive $\phi \in C_c^\infty(\mathcal{O})$ and $k \in \mathbf{R}$ we have

$$E_-(\phi(v-k)) \neq \emptyset \implies \exists y \in E_-(\phi(v-k)) : \\ H\left(y, v(y), -\frac{v(y)-k}{\phi(y)}\nabla\phi(y)\right) \geq 0$$

Finally, a function $v \in C(\mathcal{O})$ is called a **viscosity solution** of $H(y, V, \nabla V) = 0$ if for this equation it is both a viscosity subsolution and a viscosity supersolution. $\triangledown$

The beauty of viscosity solutions lies in the fact that the optimal value function is the unique viscosity solution for a given HJB equation [Crandall and Lions, 1983]. Based on the stability and regularity properties of viscosity solutions, Munos [1997] presents a convergent reinforcement learning algorithm that employs a finite-differences scheme to solve the HJB equation.

Now we turn to our second skeptical question: does this even matter in practice? We can answer this affirmitively by demonstration. We implement an algorithm like that described in Munos [1997] and compare its performance to Q-learning in this toy example. The results are shown in Figure 2.3.

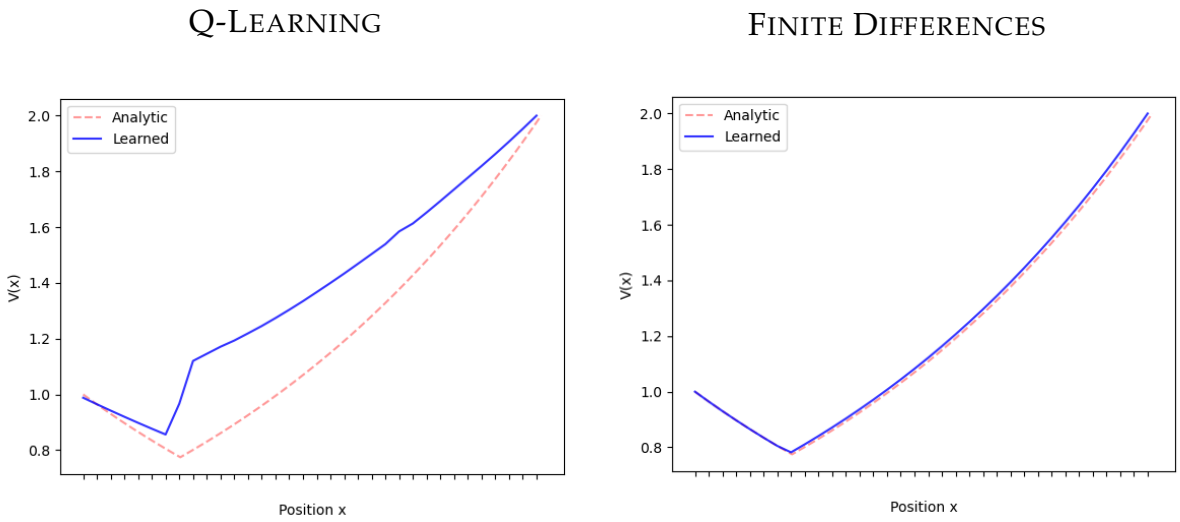


Figure 2.3: Value functions learned by a continuous-time RL algorithm and Q-Learning for Munos' toy example

In our experiments, we use a small timestep duration $\tau = 10^{-3}$ and we discretize the state space uniformly into 100 bins. Hyper-parameter search for the optimal learning rate sequence was conducted identically for both algorithms. We see that Q-learning converged to a value function with a discontinuity near the point of non-differentiability of the value function. Consequently the value function learned by Q-learning is only approximately

correct near the boundaries of the state space. On the other hand, the algorithm with continuous-time corrections learns a continuous value function that approximates the viscosity solution of (2.27) well on the entire state space.

Beyond this simple environment, Doya [2000] presents a number of algorithms that are extensions of existing (discrete-time) RL algorithms to the continuous time setting, including a method with differential updates based on the gradient of the value function, and encorporating TD(λ) updates [Sutton, 1988] to reduce bias. This algorithm was reported to have learned optimal controllers for complex control tasks faster than discrete-time counterparts by a substantial amount.

2.3.2 Unveiling Problems in Discrete Reinforcement Learning

Another reason for studying reinforcement learning in the continuous-time setting is that it forces us to recognize some of the technical challenges of reinforcement learning that are obscured, but still present, in the discrete-time MDP model. One such example is part of the credit assignment problem, which is the problem of determining which actions from a given episode in an MDP contributed most to the return. This problem is certainly addressed in the reinforcement learning literature [Sutton, 1984, Arumugam et al., 2021], however the study of reinforcement learning in continuous time demonstrates a new challenge.

The work of Baird III [1993] presents an interesting challenge for credit assignment when the timestep is small (or infinitesimal). For systems evolving continuously in time, we expect different controls to have similar Q-values at a given state – that is, individual actions should not perturb the overall return too greatly. In fact, it is shown in Baird III [1993] and formally proved in Tallec et al. [2019] that as the timestep shrinks, the Q-function may completely disregard the action altogether. In that case, Q-learning at the very least should be expected to learn extremely slowly, if at all. In order to correct this, the *Advantage Updating* algorithm is presented, where "advantage" functions are learned as opposed to Q-functions, and the advantage functions A are related to the Q-functions via

$$A(x,a) = \frac{1}{\tau}\left(Q(x,a) - \max_{a' \in \mathcal{A}} Q(x,a')\right) \tag{2.29}$$

where τ is the duration of the timestep. By effectively scaling Q-functions by τ^{-1}, the advantage functions do not lose information about actions in the continuous time limit, and Baird [1994] reports that advantage learning converged 10^5 times faster than Q-learning

in their experiments. The work of Bellemare et al. [2016] develops further on this idea by demonstrating a class of "consistent" Bellman-like operators, and empirical results in complex experiments using consistent updates substantially improve over previous similar RL algorithms.

Another approach to solving the continuous-time credit assignment problem was recently suggested by Kim et al. [2021], which presents the Hamilton-Jacobi Deep Q Network (HJ-DQN). This framework allows us to perform value-based RL when the action space is continuous, under the assumption that the control signal is Lipschitz-continuous in time. This assumption effectively means that the control is not changing too quickly, so the individual controls have more influence on the final return. Moreover, by the definition of Lipschitz-continuity, the time-derivative of the control signal is bounded by some constant L and we can determine the optimal action at each timestep (even though there are infinitely many candidates!) by

$$\dot{a}(t) = L\frac{\nabla_a Q(x,a)}{\|\nabla_a Q(x,a)\|}$$

Simply put – we must determine the direction in action space that maximizes the Q-function most quickly by computing the a-gradient of the Q function, and then perturb the control as much as we're allowed to by the Lipschitz constraint in that direction.

A similar feat can be accomplished if we make an alternative assumption that the dynamics are *control-affine*, meaning $\dot{x}(t) = f(x(t)) + \langle g, a(t)\rangle$ for some linear operator g and arbitrary function f. In particular, under this class of dynamics, value-maximizing actions can be computed in closed form even when the action space is uncountable [Tassa and Erez, 2007]. Recently, this idea was applied with great success in the Continuous Fitted Value Iteration (cFVI) algorithm [Lutter et al., 2021b] and in the Robust Fitted Value Iteration (rFVI) algorithm [Lutter et al., 2021a].

The study of Tallec et al. [2019] also discusses what is a seemingly un-explored parameter of reinforcement learning models: the length of the timestep. Even when the timesteps are not miniscule, it is not unreasonable to suspect that RL algorithms might perform differently due to changes in the timestep length. In fact, their work shows that existing discrete-time value-based RL algorithms are quite sensitive to the time discretization parameter, while continuous-time RL algorithms tend to be much more stable. This on its own demonstrates the importance of studying RL in the continuous-time limit.

2.3.3 The Stochastic Case

Continuous-time optimal control of a system governed by stochastic dynamics has been thoroughly studied [Fleming and Soner, 2006]. In most of the literature, the Markov processes underlying the evolution of the system are assumed to be Itô Diffusions (shown in §2.2.2), and this will be the case in the remainder of this thesis as well.

In the stochastic control setting, the HJB equation is not quite the same as (2.27) – this is mainly due to the extra second-order term that emerges by Itô's lemma, shown in (C.2). Deriving the corresponding equation, however, can be done trivially by exploiting the Feynman-Kac formula, shown in Appendix C.3. Simply by observation, for any fixed policy π we have

$$V^\pi(x) = \mathbf{E}\left[\int_0^\infty \gamma^t r(X_t, A_t)dt \;\middle|\; X_0 = x, A_t \sim \pi(\cdot \mid X_t)\right]$$

According to the Feynman-Kac formula, this is the exact form of the solution to the PDE

$$0 = \langle \nabla_x V^\pi(x), f_\pi(x)\rangle + \frac{1}{2}\,\mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \mathsf{H}_x V^\pi(x)\boldsymbol{\sigma}_\pi(x)\right) + V^\pi(x)\log\gamma \tag{2.30}$$

where the state process $(X_t)_{t\geq 0}$ is governed by the Itô diffusion

$$dX_t = f_\pi(X_t, A_t)dt + \boldsymbol{\sigma}_\pi(X_t)dB_t.$$

Existing continuous-time reinforcement learning algorithms have been adapted to account for (2.30). A great example is the algorithm introduced in Munos and Bourgine [1997], which extends the finite differences algorithm from Munos [2004] by deriving a finite differences scheme to account for the second order term $\boldsymbol{\sigma}_\pi(X_t)^\top \mathsf{H}_x V^\pi(x)\boldsymbol{\sigma}_\pi(X_t)$. More recently, stochastic control algorithms with powerful function approximation have been studied, using techniques such as the construction of forward-backward SDEs to allow for efficient backpropagation [Pereira et al., 2019] and importance sampling via Girsanov's theorem to accounting for off-policy learning [Exarchos and Theodorou, 2018].

2.4 Distributional Reinforcement Learning

In many applications of machine learning, it is desirable to understand the uncertainty involved in predictions from a model. For instance, in robotics applications it is usually a good idea to use knowledge of uncertainty to provide safety margins in order to prevent serious damages, and in clustering algorithms we can assign to each datapoint a distribution over possible categories. The young field of distributional reinforcement learning takes uncertainty modeling to the core object of interest in value-based RL: the value function itself.

More formally, the idea behind distributional RL is to model the probability distribution of random returns as opposed to just its expected value. While this idea may seem innocuous at first glance, estimating return distributions presents a plethora of mathematical challenges.

Recall that a popular technique for learning the value function in RL is to derive a contractive operator on the space of value functions and invoke the Banach fixed-point theorem to prove that repeated applications of this operator will yield the value function. Already, to bridge this technique to the distributional framework, we are immediately faced with problems that must be addressed:

1. Can the return distribution function be expressed as the fixed point of an operator?
2. What is a contraction on the space of probability measures, and in particular, what does it mean for probability measures to be close to each other?
3. What is an *optimal* return distribution function?

In the seminal work of Bellemare et al. [2017a], some of these questions are answered. By extending the results concerning analysis of fixed points on the space of distributions due to Rösler [1992], Bellemare et al. [2017a] shows that the return distribution does indeed satisfy a fixed-point equation for a distributional extension of the Bellman operator. However, in order to satisfy a fixed point equation of any kind, we must be clear about the metric space that we are analyzing. As it turns out, the distributional Bellman operator is not contractive for several familiar topologies on the space of probability measures [Bellemare et al., 2017a], such as the total variation distance and the Kullback-Leibler (KL) divergence[9].

An illustrative example of this, inspired by an example given by Professor Prakash Panan-

[9]The KL divergence is actually not a metric, as it is not symmetric. However, there are many ways to construct metrics out of the KL divergence that preserve its properties.

gaden in a talk at Mila, is as follows. Suppose there exists a state x in an MDP for which the return is deterministically some number y, so its return distribution is $\eta(\cdot \mid x) = \delta_y$. Let's say we're estimating this distribution by a distribution $\mu(\cdot \mid x) = \delta_z$. As long as $z \neq y$, the total variation distance between $\mu(\cdot \mid x)$ and $\eta(\cdot \mid x)$ is given by

$$\mathrm{TV}(\mu, \eta) \triangleq \frac{1}{2} \sup_{A \in \mathcal{F}} |\mu(A \mid x) - \eta(A \mid x)| = 1$$

where $\mathcal{F}$ is the σ-algebra associated with the measurable space of interest. Notably, regardless of how far z is from y (note that this notion of distance is familiar, since $y, z \in \mathbf{R}$), the learning signal according to the total variation distance is always 1! That is, total variation distance would tell us that the probability distributions $\delta_{1.0001}$ and δ_{10^7} are equidistant from δ_1, for example. Clearly, this does not capture the notion of similarity between probability distributions that we expect in reinforcement learning.

An insight from the previous example is that the notion of distance between probability measures in $\mathcal{P}_p(\mathcal{W})$ should be related in some way to the topology of $\mathcal{W}$. This is captured nicely by the *Wasserstein distances* [Villani, 2008].

Definition 6 (Wasserstein distance)**.** Let $(\mathcal{W}, d)$ be a metric space and $(\mathcal{W}, \mathcal{F})$ a measurable space over which μ, ν are measures. A (probabilistic) **coupling** between μ, ν is a probability measure π on the product space $\mathcal{W} \times \mathcal{W}$ such that $\mu = (\mathsf{id}, \mathcal{W})_\sharp \pi$ and $\nu = (\mathcal{X}, \mathsf{id})_\sharp \pi$ – that is, the marginals of π are μ, ν respectively.

Suppose $\mathcal{W}$ is a normed space, and let Π denote the set of all couplings of measures in $\mathcal{P}_p(\mathcal{W})$. For $p \in \{1, \ldots, \}$, the p-**Wasserstein distance** $d_{\mathbf{W}_p}$ is defined as

$$d_{\mathbf{W}_p}(\mu, \nu) = \min_{\pi \in \Pi} \left(\int_{\mathcal{W} \times \mathcal{W}} \|x - y\|^p d\pi(x, y) \right)^{1/p}$$

Note that the "optimal coupling" satisfying the minimization above always exists, and $d_{\mathbf{W}_p}$ is a metric over $\mathcal{P}_p(\mathcal{W})$ [Villani, 2008].

The metric space $(\mathcal{P}_p(\mathcal{W}), d_{\mathbf{W}_p})$ is denoted by $\mathbf{W}_p(\mathcal{W})$. ▽

In the notation of Rowland et al. [2019], Bellemare et al. [2017a] shows that the *distributional* Bellman operator $\mathcal{T}^\pi : \mathcal{P}_p(\mathbf{R}) \to \mathcal{P}_p(\mathbf{R})$ defined as

$$\mathscr{T}^{\pi}\,\eta(\cdot \mid x, a) = \mathop{\mathbf{E}}_{\substack{x' \sim P(\cdot|x,a) \\ a' \sim \pi(\cdot|x')}} \left[{f^{r,\gamma}}_{\sharp}\,\eta(\cdot \mid x', a') \right]$$
$$f^{r,\gamma}(\xi) = r + \gamma\xi,$$

is a contraction in a "supremal form" $\overline{d}_{\mathbf{W}_p}$ of the p-Wasserstein metric,

$$\overline{d}_{\mathbf{W}_p}(\mu, \nu) = \sup_{\substack{x \in \mathcal{X} \\ a \in \mathcal{A}}} d_{\mathbf{W}_p}(\mu(\cdot \mid x, a), \nu(\cdot \mid x, a)).$$

In the same work, it is shown that $\overline{d}_{\mathbf{W}_p}$ is indeed a metric on the space of functions with the type signature $\mathcal{X} \times \mathcal{A} \to \mathcal{P}_p(\mathbf{R})$.

At the time, there was no known, tractable method for computing gradients of Wasserstein distances from samples [Bellemare et al., 2017b], so consequently the C51 algorithm presented in Bellemare et al. [2017a] worked by backpropagating gradients of the KL divergence, despite the negative theoretical results. C51 was tremendously successful, and to this day it contributes positively to state of the art deep reinforcement learning models [Hessel et al., 2018].

Soon after, Dabney et al. [2018a] discovered a method for minimizing the 1-Wasserstein metric from samples by exploiting a property of Wasserstein distances over the reals [Thorpe, 2018, Theorem 2.1] and performing quantile regression [Koenker and Bassett Jr, 1978]. This resulted in another exceptional deep RL algorithm, known as QR-DQN, which also maintains its presence in state of the art models [Bellemare et al., 2020].

Following this, Rowland et al. [2019] constructs a framework for proper return distribution learning by distinguishing return distribution samples from return distribution statistics. In this work, methods of representing probability measures as functions of their statistics are characterized according to how well they can approximate fixed points of certain distributional operators, including the distributional Bellman operator. This analysis partly explains the great empirical performance of C51 given its deviation from the theory of distributional RL.

There have since been many developments in distributional RL concerning representations of the return distributions, as will be discussed later in §4.1. Additionally, distribu-

tional RL has been studied as a tool for promoting exploration in RL [Mavrin et al., 2019] as well as *safe* exploration [Zhang and Weng, 2021].

Aside from the contributions of this thesis, distributional RL in continuous time has only been studied by the concurrent work of Halperin [2021], which focuses mainly on risk-sensitive RL in continuous time and provides no algorithms for learning return distribution functions.

2.5 Gradient Flows in Abstract Metric Spaces

This section will delve into a formalism for studying certain iterative refinement algorithms in the limit as the iterations occur continuously in time. In Chapter 4, we will use this formalism to describe a continuous-time extension of policy evaluation. We will make extensive use of the topological concepts of Appendix A, particularly §A.1 where metric spaces are defined. More in-depth treatments of the topics in this section can be found in Ambrosio et al. [2008], Santambrogio [2015], and Villani [2008].

As we saw in §2.1.3, a common approach to solving the Bellman equation is by iteratively applying an operator to an initial guess of the value function until a fixed point is reached. However, such an iterative method is a discrete-time operation by its very nature. When developing continuous-time RL algorithms later in the thesis, we will look at the "continuous-time" limit of such an iterative algorithm. Purely for the sake of building intuition, we can think of this continuous-time limit as a solution to the *Cauchy problem*[Ambrosio et al., 2008],

$$\begin{cases} \frac{\partial}{\partial t} v(t,\cdot) = -\nabla \mathscr{F}(v(t,\cdot)) \\ v(0,x) = V_0(x) \end{cases} \tag{2.31}$$

where $v(t,\cdot) \in \mathcal{V}$ represents the estimate of the value function at time t among the class of value functions $\mathcal{V}$, $\mathscr{F} : \mathcal{V} \to \mathbf{R}$ is a *loss functional*, and $V_0 \in \mathcal{V}$ is the initial guess of the value function. In Q-Learning, we update estimates of the Q-function so as to minimize the squared error between Q and $\mathscr{T}Q$, so we may interpret (2.31) as a continuous-time process during which the value function $v(t,\cdot)$ moves in the direction of the steepest descent of the error signal it incurs. More succinctly, this represents a continuous-time extension of gradient descent, which is called a *gradient flow*.

However, in a general metric space, the Cauchy problem has no meaning, and conse-

quently we must consider an alternative formulation in order to make sense of the Cauchy problem in spaces without much algebraic structure. If we look more closely at (2.31), we note that neither of its two terms have any proper meaning when $\mathcal{V}$ is an arbitrary metric space [Ambrosio et al., 2008]. Indeed, the familiar definition of the time derivative would be expressed as $\frac{\partial}{\partial t}v(t,\cdot) = \lim_{\delta\to 0}\frac{1}{\delta}(v(t+\delta,\cdot) - v(t,\cdot))$, and this requires that $\mathcal{V}$ is closed under an invertible, associative binary operation as well as scalar multiplication (i.e, $\mathcal{V}$ should be a vector space). This condition of course is not satisfied when it is only assumed that $\mathcal{V}$ is a metric space.

While such an issue may seem like an unnecessary technicality, it most certainly is not. A relevant example that demonstrates why this abstraction is necessary is when $\mathcal{V}$ is a space of return distribution functions. One must be very careful when performing algebraic operations on such objects – most often return distribution functions *do not* form a vector space. Suppose $\eta \in \mathcal{V}$ and $|\alpha| \neq 1$. Then since $\eta(\mathcal{R} \mid x) = 1$ by definition, we must have $\alpha\eta(\mathcal{R} \mid x) \neq 1$, so $\alpha\eta$ is not a probability measure, and it follows that $\mathcal{V}$ is not closed under scalar multiplication. Thus, certainly for the purpose of this thesis, we must study an abstraction of gradient flows to general metric spaces.

2.5.1 Evolution Variational Inequality

A clever way to deal with generalizing the gradient flow formulation involves expressing the gradient flow in a simple metric space (say, a Hilbert space) by an equivalent identity comprised solely of metric operations. Since the resulting expression is equivalent to a gradient flow, it provides a meaningful notion of a gradient flow in spaces that don't necessary have a vector space structure. Generally, this requires invoking assumptions on $\mathscr{F}$.

A very useful characterization is known as an *evolution variational inequality* [De Giorgi et al., 1980]. We begin by assuming that $\mathcal{V}$ is a Hilbert space and $\mathscr{F}$ is λ-convex [Santambrogio, 2015], meaning that for every $\psi \in \mathcal{V}$, we have

$$\mathscr{F}(\psi) \geq \mathscr{F}(\phi(t)) + \frac{\lambda}{2}\|\phi(t) - \psi\|^2 + \langle p, \psi - \phi(t)\rangle$$

where $\phi : \mathbf{R}_+ \to \mathcal{V}$ and p is in the subdifferential of $\mathscr{F}$ evaluated at $\phi(t)$. If ϕ is a solution to the Cauchy problem (2.31), then $p = -\frac{\partial}{\partial t}\phi(t)$. It follows that

$$\mathscr{F}(\psi) \geq \mathscr{F}(\phi(t)) + \frac{\lambda}{2}\langle \phi(t) - \psi, \phi(t) - \psi \rangle - \langle \phi'(t), \psi - \phi(t) \rangle$$

Moreover, note that

$$\begin{aligned} \frac{1}{2}\frac{\partial}{\partial t}\|\phi(t) - \psi\|^2 &= \left\langle \frac{\partial}{\partial t}(\phi(t) - \psi), \phi(t) - \psi \right\rangle \\ &= \left\langle \frac{\partial}{\partial t}\phi(t), \phi(t) - \psi \right\rangle \end{aligned}$$

So, when $\mathscr{F}$ is λ-convex and ϕ solves the Cauchy problem, we have

$$\mathscr{F}(\psi) \geq \mathscr{F}(\phi(t)) + \frac{\lambda}{2}\|\phi(t) - \psi\|^2 + \frac{1}{2}\frac{\partial}{\partial t}\|\phi(t) - \psi\|^2$$

Finally, noting that $\|x - y\|^2 = d^2(x, y)$ where d is the metric in the Hilbert space $\mathcal{V}$, we have

$$\frac{1}{2}d^2(\phi(t), \psi) \leq \mathscr{F}(\psi) - \mathscr{F}(\phi(t)) - \frac{\lambda}{2}d^2(\phi(t), \psi) \tag{2.32}$$

Equation (2.32) is known as the EVI_λ inequality. Note that this inequality is expressed only in terms of metric quantities, so it is a suitable characterization of a gradient flow in abstract metric space as long as the concept of λ-convexity can also be defined in terms of metric quantities. Fortunately, it is known [Muratori and Savaré, 2018] that $\mathscr{F}$ is λ-convex if and only if for every $\phi, \psi \in \mathcal{V}$ there exists a geodesic[10] $(\varrho_t)_{t\in[0,1]}$ with $\varrho_0 = \phi$ and $\varrho_1 = \psi$ such that

[10] A geodesic between two points is a curve between those points for which the arc length of the curve according to the space's metric is minimal.

$$\mathscr{F}(\varrho_t) \leq (1-t)\mathscr{F}(\phi) + t\mathscr{F}(\psi) - \frac{\lambda}{2}t(1-t)d^2(\phi,\psi) \tag{2.33}$$

Note that when $\lambda > 0$, which we generally desire, λ-convexity is a stronger property than convexity. Hence, we will consider the following characterization of gradient flows.

Definition 7 (Abstract Gradient Flow)**.** Let $(\mathcal{V}, d)$ be a metric space, and let $\phi : \mathbf{R}_+ \to \mathcal{V}$ be a curve in $\mathcal{V}$. If $\mathcal{F}$ is λ-convex in the sense of (2.33) and ϕ satisties the EVI_λ inequality (2.32), then ϕ is said to be a (EVI-type) **gradient flow** of $\mathscr{F}$. ▽

An attractive property of EVI-type gradient flows is that they satisfy a contraction property, which will play an analogous role to contraction mappings in discrete-time analysis. This suggests that EVI-type gradient flows are a promising candidate for describing continuous-time policy evaluation.

Theorem 2.5 (Uniqueness of Gradient Flows, [Santambrogio, 2015])**.** *Let $(\mathcal{V}, d)$ be a metric space. If two curves $\phi, \varphi : \mathbf{R}_+ \to \mathcal{V}$ satisfy the EVI_λ inequality (2.32) for some $\lambda \geq 0$ and a λ-convex functional $\mathscr{F}$, then*

$$\frac{d}{dt}d^2(\phi(t), \varphi(t)) \leq -2\lambda d(\phi(t), \varphi(t))$$

Then, by Grönwall's lemma [Gronwall, 1919], it follows that

$$d(\phi(t), \varphi(t)) \leq e^{-\lambda t} d(\phi(0), \varphi(0))$$

This shows that any two EVI-type gradient flows for a common λ-convex loss functional will eventually coincide. Consequently, if continuous-time policy evaluation can be framed as an EVI-type gradient flow, we can be assured that the continuous-time policy evaluation process will converge to a unique fixed point (say, the return distribution function). The work of Martin et al. [2019] exploits this concept to formulate distributional policy evaluation of a discrete-time process as an EVI-type gradient flow, and we will develop this idea further in Chapter 4 for continuous-time policy evaluation.

2.5.2 Wasserstein Gradient Flows

In §2.4 we defined the Wasserstein distance as a convenient distance measurement between probability measures in distributional RL. It is a beautiful result from optimal

transport theory that the Wasserstein distances are proper metrics over spaces of probability measures [Villani, 2008]. For a metric space $(\mathcal{V}, d)$, the metric space $(\mathcal{P}_p(\mathcal{V}), d_{\mathbf{W}_p})$ (with $d_{\mathbf{W}_p}$ having the same definition as in definition 6) is called the p-Wasserstein space, and it is denoted by $\mathbf{W}_p$.

Among all Wasserstein spaces, the space $\mathbf{W}_2$ is by and large the most convenient for the analysis of smooth curves in the space of probability measures [Santambrogio, 2016]. For this reason, the analysis of gradient flows in probability measure space is often conducted in $\mathbf{W}_2$, and the term *Wasserstein Gradient Flow* (WGF) almost exclusively refers to gradient flows in $\mathbf{W}_2$ specifically.

Perhaps the most celebrated result in the study of WGFs is that of Jordan, Kinderlehrer, and Otto [Jordan et al., 2002], known as the JKO scheme. As an overview, the result is comprised of the following points:

1. The **Fokker-Planck** equation, which is a widely studied PDE in physics and is given by

$$\frac{\partial}{\partial t}\varrho(t,x) = -\frac{\partial}{\partial x}\left(\mu(x,t)\varrho(x,t)\right) + \frac{\partial^2}{\partial x^2}\left(\sigma^2(t,x)\varrho(t,x)\right) \tag{FP}$$

   where μ, σ are known and σ is positive definite, satisfies the **continuity equation**

$$\frac{\partial}{\partial t}\varrho(t,x) + \nabla_x \cdot (\varrho(t,x)\mathbf{v}(t,x)) = 0 \tag{CE}$$

   for some vector field $\mathbf{v}$, which characterizes the conservation of mass of the process ϱ. If we think of $\varrho(t,\cdot)$ as a probability density, this means that we can interpret (FP) as an equation governing the evolution of a probability density that conserves measure.

2. The Fokker-Planck equation (FP) for a constant σ is the **Cauchy problem** associated to the **Wasserstein gradient flow** of the functional $\mathscr{F}$ given by

$$\mathscr{F}(\varrho(t,\cdot)) = \int_{\mathcal{X}} U(x)\varrho(t,dx) - \sigma\mathcal{H}(\varrho(t,\cdot)) \tag{FPWGF}$$

   where $\mathbf{v}(t,x) = -\nabla U(x)$.

3. The loss functional (FPWGF) can equivalently be written as $\mathscr{F}(\varrho(t,\cdot)) = D_{\mathrm{KL}}(\varrho(t,\cdot) \parallel \mu)$ where

$$\mu(x) = \frac{1}{Z} e^{-U(x)}$$
$$Z \triangleq \int_{\mathcal{X}} e^{-U(x)} dx$$

Thus, the Fokker-Planck equation can be interpreted as the evolution of a probability density towards μ in the sense of KL divergence. Naturally, it follows that μ is the stationary solution of (FP).

4. Following the analysis of **generalized minimizing movements schemes** [De Giorgi, 1993], it is shown that under an appropriate time discretization (discussed further in §4.2), for any timestep τ, sequences given by

$$\varrho_{k+1}^{\tau} \in \arg\min_{\nu \in \mathbf{W}_2} \left[\mathscr{F}(\nu) + \frac{1}{2\tau} d_{\mathbf{W}_2}^2(\nu, \varrho_k^{\tau}) \right]$$

satisfy $\lim_{\tau \to 0} \varrho_k^{\tau} \to \mu$, where convergence is attained with respect to $d_{\mathbf{W}_2}$. The JKO scheme refers to the process of **approximately solving** (FP) by **iteratively** computing terms of the sequence $\{\varrho_k^{\tau}\}_{k=1}^{\infty}$ for small τ, where ϱ_0^{τ} is an arbitrary probability density. Notably, this approximation converges to the true gradient flow as $\tau \to 0$.

In the context of reinforcement learning, this result is very useful, as it describes a convergent method to solve a Cauchy problem discretized in time that is similar to dynamic programming. This will be the focus of §4.2.

3

Evolution of Return Distributions

We will now shift our focus to formally representing the return distribution function for an RL agent evolving continuously in time with a fixed behavioral policy. In order to do so, it will be necessary to impose some structural and regularity properties on the dynamics of the environment and on the return distributions. More concretely, the chapter is structured as follows,

- A formalism of **continuous-time Markov processes** will be given;
- The **random return** is formulated as a **special type of Markovian process** in §3.1;
- A **distributional analogue to the HJB equation** (see equation (2.30)) is derived in §3.2.

In order to model stochastic trajectories in continuous time, we will use the language of stochastic processes and stochastic differential equations as discussed in §2.2 and Appendix C. Moreover, we must discuss what it means for a continuous-time process to be Markovian.

Definition 8 (Markov Process, Rogers and Williams [1994])**.** Let $(X_t)_{t\geq 0}$ be a stochastic process in the filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \Pr)$. A *Markovian transition kernel* is

a kernel with a continuous parameter t, $P_t : \Omega \times \mathcal{F} \to [0,1]$, such that for any bounded $\mathcal{F}$-measurable function f, we have

$$(P_t f)(X_s) = \mathbf{E}\left[f(X_{s+t}) \mid \mathcal{F}_s\right] \qquad \Pr-\text{almost surely} \tag{3.1}$$

A collection $(P_t)_{t\geq 0}$ of Markovian transition kernels is called a *transition semigroup*[1] when

1. For each $t \geq 0$ and $x \in \Omega$, $P_t(x, \cdot)$ is a measure on $\mathcal{F}$ and $P_t(x, \Omega) \leq 1$;
2. For each $t \geq 0$ and $\Gamma \in \mathcal{F}$, the mapping $P_t(\cdot, \Gamma)$ is $\mathcal{F}$-measurable; and
3. (The Chapman-Kolmogorov Identity) For each $s, t \geq 0$,each $x \in \Omega$, and each $\Gamma \in \mathcal{F}$, the collection satisfies

$$P_{s+t}(x, \Gamma) = \int_\Omega P_s(x, dy) P_t(y, \Gamma)$$

Then $P_t P_s = P_{t+s}$, so $(P_t)_{t\geq 0}$ is indeed a semigroup.

A *Markov process* is a stochastic process $(X_t)_{t\geq 0}$ together with a transition semigroup $(P_t)_{t\geq 0}$ such that (3.1) holds. ▽

Beyond the Markovian property, we will further require that the trajectory of the agent is "regular enough" for us to study its instantaneous dynamics. In particular, we will assume henceforth that the trajectory of the agent is a *Feller-Dynkin* process.

Definition 9 (Feller-Dynkin Process, Infinitesimal Generator, Rogers and Williams [1994])**.** Consider a filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \Pr)$ and let $\mathcal{X}$ be a Polish[2] space. A transition semigroup $(P_t)_{t\geq 0}$ is said to be a *Feller semigroup* if

1. $P_t : C_0(\mathcal{X}) \to C_0(\mathcal{X})$ for each $t \in \mathbf{R}_+$;
2. For any $f \in C_0(\mathcal{X})$ with $f \leq 1$, $P_t f \in [0,1]$;
3. $P_s P_t = P_{s+t}$ and $P_0 = \mathsf{id}$;
4. For any $f \in C_0(\mathcal{X})$, we have $\|P_t f - f\| \xrightarrow{t\downarrow 0} 0$.

A Markov process with a Feller semigroup is called a *Feller-Dynkin process*.

[1] This name emphasizes the semigroup nature of the collection of transition kernels. In the abstract algebra literature, a semigroup is a set of objects that is closed under an associative binary operation.

[2] A Polish space is a complete metric space that has a countable, dense subset.

Define the set $\mathscr{D}(\mathscr{L})$ according to

$$\mathscr{D}(\mathscr{L}) = \left\{ f \in C_0(\mathcal{X}) \mid \exists g \in C_0(\mathcal{X}) \quad \text{such that} \quad \|\delta^{-1}(P_\delta - f) - g\| \xrightarrow{\delta \downarrow 0} 0 \right\}$$

The *infinitesimal generator* of a Feller-Dynkin process is the operator $\mathscr{L} : \mathscr{D}(\mathscr{L}) \to C_0(E)$ where

$$\mathscr{L} f = \lim_{\delta \to 0} \frac{P_\delta f - f}{\delta}$$

and $\mathscr{D}(\mathscr{L})$ is called the *domain of the infinitesimal generator* $\mathscr{L}$. ▽

Remark 3.1. Note that Itô diffusions with Lipschitz-continuous coefficients are Feller-Dynkin processes [Le Gall, 2016].

We consider a continuous-time MDP $(\mathcal{X}, \mathcal{A}, r, (P_t), \gamma)$ where $\mathcal{X} \subset \mathbf{R}^d$ is compact, (P_t) is a Feller semigroup with infinitesimal generator $\mathscr{L}$, $r : \mathcal{X} \to \mathcal{R}_{\text{rew}} \subset \mathbf{R}$, and $\gamma \in (0, 1)$. Additionally, we impose a mild assumption on the reward function.

Assumption 3.1. The range $\mathcal{R}_{\text{rew}}$ of r is contained in an interval $[R_{\min}, R_{\max}]$, where $|R_{\min}|, |R_{\max}| < \infty$.

When assumption 3.1 is satisfied, we make the following observations regarding the extrema of the return,

$$\begin{aligned}
\mathscr{J}(x) &= \int_0^\infty \gamma^t r(X_t) dt \;\Big|\; X_0 = x \\
V_{\min} \triangleq \inf \mathscr{J}(x) &\geq \int_0^\infty \gamma^t R_{\min} dt \\
&= \frac{1}{\log \frac{1}{\gamma}} R_{\min} \\
V_{\max} \triangleq \sup \mathscr{J}(x) &\leq \int_0^\infty \gamma^t R_{\max} dt \\
&= \frac{1}{\log \frac{1}{\gamma}} R_{\max}
\end{aligned}$$

This confirms that the discounted return will be bounded. We refer to the set $\mathcal{R} = [V_{\min}, V_{\max}]$ as the *return space*.

In order to give a formal treatment of the stochastic processes generated by the agent interacting with its environment, we must specify a filtration. In particular, we will be interested for the most part in the **canonical filtration**. The canonical filtration is the filtration $(\mathcal{F}_t)_{t\geq 0}$ where $\mathcal{F}_t$ is the sub-σ-algebra generated by the trajectory observed up to time t. Naturally, $\mathcal{F}_t \subset \mathcal{F}_{t+\delta}$ for any $\delta > 0$.

We will perform analysis of the continuous-time MDP on a filtered probability space $\mathsf{P} = (\Omega, \mathcal{F}, (\mathcal{F}_t), \Pr)$, where

- $\Omega \subset \cup_{n\geq 0}(\mathbf{R}_+ \times \mathcal{X} \times \mathcal{A} \times \mathcal{R}_{\text{rew}})^n$ is the sample space of trajectories in the MDP;
- $\mathcal{F}$ is a σ-algebra over Ω;
- $(\mathcal{F}_t)_{t\geq 0}$ is the canonical filtration.

We denote by $\eta^\pi : \mathcal{X} \to \mathcal{P}_p(\mathcal{A})$ the *return distribution function*, which is defined via

$$\text{Law}\,(G_x^\pi) = \eta^\pi(\cdot \mid x),$$

where G_x^π is the random variable representing the discounted return obtained by an agent starting at state $x \in \mathcal{X}$ and following a policy π. The objects $\eta^\pi(\cdot \mid x)$ are understood as probability measures. We will also require some assumptions on the regularity of the return distribution function, which are stated below.

Assumption 3.2. At every state $x \in \mathcal{X}$, the return distribution $\eta^\pi(\cdot \mid x)$ is absolutely continuous (as a measure over the return space) with respect to the Lebesgue measure.

Assumption 3.3. The return distribution function is twice differentiable over $\mathcal{X} \times \mathcal{R}$ almost everywhere, and its second partial derivatives are continuous almost everywhere.

Furthermore, we will occasionally want to analyze some less abstract Markov processes. In these cases, we will refer to the following assumption.

Assumption 3.4 (Diffusion dynamics)**.** The Markov process $(X_t)_{t\geq 0} \subset \mathcal{X} \subset \mathbf{R}^d$ induced by the agent following a fixed (stochastic) policy π is an Itô diffusion evolving according to

$$dX_t = f_\pi(X_t)dt + \boldsymbol{\sigma}_\pi(X_t)dB_t \tag{3.2}$$

where $f_\pi : \mathcal{X} \to \mathcal{X}$, $\boldsymbol{\sigma}_\pi : \mathcal{X} \to \mathbf{R}^{d\times d}$ are Lipschitz-continuous, $\boldsymbol{\sigma}_\pi$ is positive semidefinite,

and B_t is a Brownian motion.

3.1 The Stochastic Process of Truncated Returns

We would like to understand how estimates of the random return should evolve over time. Unfortunately, a function mapping states to (random) returns cannot be progressively measurable, as it requires knowledge of an entire trajectory to be computed. Therefore, we will not be able to study random returns directly with the machinery of stochastic calculus. Instead, we'll introduce another stochastic process as a "gateway" to the random return.

Definition 10 (The Truncated Return Process)**.** Let $(\mathcal{X}, \mathcal{A}, r, (P_t), \gamma)$ be a continuous-time MDP. The *truncated return process* is a stochastic process $(J_t)_{t\geq 0} \in \mathcal{X} \times \mathcal{R}$ given by

$$J_t = (X_t, \overline{G}_t) \qquad \overline{G}_t = \int_0^t \gamma^s r(X_s) ds$$

The values $\overline{G}_t$ are simply the discounted rewards accumulated up to time t, and $\overline{G}_0 \equiv 0$. $\triangledown$

Proposition 1. *The truncated return process $(J_t)_{t\geq 0}$ is a Markov process with respect to the canonical filtration.*

Proof. Let $\psi \in C(\mathcal{X} \times \mathcal{R}; \mathbf{R})$ and $h > 0$. As usual, we denote the canonical filtration by $(\mathcal{F}_t)_{t\geq 0}$. By the definition of the truncated return process,

$$\begin{aligned}
\mathbf{E}\left[\psi(J_{t+h}) \mid \mathcal{F}_t\right] &= \mathbf{E}\left[\psi(X_{t+h}, \overline{G}_{t+h}) \;\middle|\; \mathcal{F}_t\right] \\
&= \mathbf{E}\left[\psi\left(X_{t+h}, \overline{G}_t + \int_t^{t+h} \gamma^s r(X_s) ds\right) \;\middle|\; \mathcal{F}_t\right] \\
&= \mathbf{E}\left[\psi\left(X_{t+h}, \overline{G}_t + \int_t^{t+h} \gamma^s r(X_s) ds\right) \;\middle|\; J_t\right]
\end{aligned}$$

where the final step holds since the process $(X_t)_{t\geq 0}$ is assumed to be Markovian. Thus, we've shown that for any $\psi \in C(\mathcal{X} \times \mathcal{R}; \mathbf{R})$, there exists a function $m : \mathcal{X} \times \mathcal{R} \to \mathbf{R}$ where

$$\mathbf{E}\left[\psi(J_{t+h}) \mid \mathcal{F}_t\right] = m(X_t, \overline{G}_t)$$

Therefore, the process $(J_t)_{t\geq 0}$ is Markovian. $\square$

It will be helpful to think of the random return in terms of the truncated return process. To do so, we'll need to formalize the concept of a trajectory terminating at a non-deterministic time.

Definition 11 (Stopping time, [Le Gall, 2016])**.** Let $(\Omega, \mathcal{F}, (\mathcal{F}_t))$ be a measurable space with filtration $(\mathcal{F}_t)$. A random variable $T : \Omega \to \mathbf{R}_+$ is called a *stopping time* with respect to the filtration $(\mathcal{F}_t)$ if

$$\{T \leq t\} \in \mathcal{F}_t \qquad t \geq 0$$

We define the *σ-algebra of the past before T* as the σ-algebra $\mathcal{F}_T$ given by

$$\mathcal{F}_T = \{A \in \mathcal{F}_\infty : A \cap \{T \leq t\} \in \mathcal{F}_t\}$$

$\triangledown$

Since trajectories are assumed to be Markovian, it is natural to expect their termination to occur once the agent has reached a state from some set of *terminating states*.

Assumption 3.5 (Terminating states)**.** The continuous-time MDP $(\mathcal{X}, \mathcal{A}, r, (P_t)_{t\geq 0}, \gamma)$ admits a measurable set $\mathcal{G} \subset \mathcal{X}$, referred to as the *terminating states*, such that trajectories terminate when the agent reaches any state $x \in \mathcal{G}$.

We will confirm that the random termination time corresponding to the first entry of $(X_t)_{t\geq 0}$ into $\mathcal{G}$ is a stopping time.

Proposition 2. *Consider a filtered probability space $(\Omega, \mathcal{F}, \Pr)$. Let T denote the first time that an agent enters a state among a fixed measurable set of terminating states $\mathcal{G}$, so*

$$T = \inf_{t\geq 0}\{X_t \in \mathcal{G}\}$$

Then if $\mu(T < \infty) = 1$, T is a stopping time with respect to the canonical filtration.

Proof. The proof is simple. For any $\epsilon > 0$, there exists $t' \in \mathbf{R}$ such that $\Pr(T > t') \leq \epsilon$. Thus, with probability at least $1 - \epsilon$, T lies in the compact set $[0, t']$. Therefore, the function $t \mapsto t\mathbf{1}_{[X_t \in \mathcal{G}]}$ almost surely attains its infimum. Since the characteristic function $\omega \mapsto \chi_{\mathcal{G}}(X_t(\omega)) = \mathbf{1}_{[X_t(\omega)\in\mathcal{G}]}$ is $\mathcal{F}_t$-measurable, it follows that $\inf_{t\geq 0}\{T \leq t\} \in \mathcal{F}_t$, so T is a stopping time. $\square$

In the remainder of the thesis, we will be interested in the random (discounted) return G_x^π starting at a state $x \in \mathcal{X}$ and following the policy π. G_x^π is a random variable due to the

fact that the state transitions are random. We define it as follows,

$$G_x^\pi = \int_0^T \gamma^t r(X_t) dt \;\Big|\; X_0 = x \tag{3.3}$$

It's clear that

$$\overline{G}_T \overset{\mathcal{L}}{=} G_x^\pi \;\Big|\; X_0 = x$$

The reason for studying the process $(\overline{G}_t)_{t\geq 0}$ as opposed to G_x^π is that $(\overline{G}_t)_{t\geq 0}$ is adapted to the canonical filtration, whereas G_x^π is only measurable with respect to $\mathcal{F}_\infty$.

In temporal difference learning, we perform approximate dynamic programming to solve the Bellman equation by using the difference between the value function at a given state and the estimated value bootstrapped by the value function at the next state as a learning signal. However, in continuous time, the notion of a "next state" is meaningless. Instead, we study the rate of change of the value function and approximately solve the resulting PDE. This leaves another glaring question though: how should one measure or interpret the rate of change of a noisy (stochastic) signal? To answer this, we must first introduce some regularity conditions on the dynamics of the stochastic processes in question.

The following theorem, due to Kolmogoroff [1931], will be instrumental in the sequel. A proof is given for clarity.

Theorem 3.1 (Kolmogorov Backward Equation)**.** *Let $(X_t)_{t\geq 0} \subset \overline{\mathcal{O}}$ be a Feller-Dynkin process for a metric space $\mathcal{O} \subset \mathcal{X}$ and consider the probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t), \Pr)$. Denote by T the infimum over times t for which $X_t \notin \mathcal{O}$. For any measurable function ϕ that is absolutely continuous and differentiable almost everywhere, the function $u(x,s) = \mathbf{E}[\phi(X_T) \mid X_{s\wedge T} = x]$ solves the PDE*

$$\frac{\partial u(x,s)}{\partial s} = -\mathscr{L} u(x,s) \tag{3.4}$$

with the terminal condition $u(x,t) = \phi(x)$ when $x \in \overline{\mathcal{O}} \backslash \mathcal{O}$, where $\mathscr{L}$ is the infinitesimal generator of the process $(X_t)_{t\geq 0}$.

In order to prove Theorem 3.1, the following lemma will be handy.

Lemma 3.1 ([Le Gall, 2016], Theorem 6.14)**.** *Let $(X_t)_{t\geq 0}$ be a Feller-Dynkin process on a metric space $\mathcal{X}$, and consider functions $h, g \in C_0(\mathcal{X})$. The following two conditions are equivalent:*

1. *$h \in \mathscr{D}(\mathscr{L})$ and $\mathscr{L}h = g$;*
2. *For each $x \in \mathcal{X}$, the process*

$$h(X_t) - \int_0^t g(X_s)ds \;\Big|\; X_0 = x$$

is a martingale with respect to the filtration $(\mathcal{F}_t)$.

Proof of Theorem 3.1. By Lemma 3.1, we know that the process $\Phi_t = \phi(X_t) - \int_0^t g(X_s)ds$ is a martingale with respect to $(\mathcal{F}_t)$. Let $s < t < T$. By the definition of a martingale, we have

$$0 = \mathbf{E}\left[\Phi_T \mid \mathcal{F}_t\right] - \mathbf{E}\left[\Phi_T \mid \mathcal{F}_s\right] = \mathbf{E}\left[h(X_T) + \int_0^T g(X_r)dr \;\middle|\; \mathcal{F}_t\right] - \mathbf{E}\left[h(X_T) + \int_0^T g(X_r)dr \;\middle|\; \mathcal{F}_s\right]$$

$$\mathbf{E}\left[h(X_T) \mid \mathcal{F}_t\right] - \mathbf{E}\left[h(X_T) \mid \mathcal{F}_s\right] = \mathbf{E}\left[\int_s^t \mathscr{L}h(X_r)dr \;\middle|\; \mathcal{F}_t\right]$$

Dividing through by $t - s$ and taking the limit as $s \uparrow t$,

$$\begin{aligned}
\frac{\partial}{\partial s}\mathbf{E}\left[\phi(X_T) \mid \mathcal{F}_s\right] = \frac{\partial}{\partial s}u(x,s) &\overset{(a)}{=} \mathbf{E}\left[\frac{\partial}{\partial s}\int_s^t \mathscr{L}\phi(X_r)dr \;\middle|\; \mathcal{F}_t\right] \\
&= -\mathbf{E}\left[\mathscr{L}\phi(X_r)dr \mid \mathcal{F}_s\right] \\
&\overset{(b)}{=} -\mathscr{L}\mathbf{E}\left[\phi(X_s) \mid \mathcal{F}_s\right] \\
&= -\mathscr{L}u(x,s)
\end{aligned}$$

Step (a) is allowed by the Leibniz integration rule since the infinitesimal generator preserves continuity and ϕ is absolutely continuous by assumption. Finally, step (b) is allowed by the linearity of expectation, since $\mathscr{L}$ is a linear operator. □

Of particular interest is the case where $\phi(\xi; A) = \chi_A(\xi)$ for any Borel set A, where $\chi_A(a) = \mathbf{1}_{[a \in A]}$ is the characteristic function for A. With our truncated return process $(J_t)_{t \geq 0}$, we have

$$u_t(z; A) \triangleq \mathbf{E}\left[\chi_A(\overline{G}_T) \;\middle|\; \mathcal{F}_t\right] = \Pr(G_T \in A \mid X_t = x, \overline{G}_t = \xi) \qquad z = (x, \xi) \tag{3.5}$$

Since $J_T = (X_T, \overline{G}_T)$ and $G_T = G_x^\pi$ is understood to be the "truncated"[3] return at the

[3]Of course, since $\overline{G}_T$ is the discounted return at the end of the episode, nothing is actually truncated.

end of a rollout, u can be interpreted as the probability measure over returns starting at a given state!

3.2 A Characterization of the Return Distributions

We're now ready to demonstrate that the return distribution function can be expressed as a solution to a Kolmogorov backward equation.

Theorem 3.2 (Distributional HJB Equation for Policy Evaluation)**.** *Let $(\mathcal{X}, \mathcal{A}, r, (P_t)_{t\geq 0}, \gamma)$ be a continuous-time MDP on which a truncated return process $(J_t)_{t\geq 0}$ is generated by following a policy π. Suppose $(X_t)_{t\geq 0} = (\iota_1 J_t)_{t\geq 0}$ is a Feller-Dynkin process, and denote its infinitesimal generator by $\mathscr{L}_X$. Recall that the return distribution function is defined such that*

$$G_x^\pi \sim \eta^\pi(\cdot \mid x)$$

where G_x^π is the random return as defined by (3.3), and suppose Assumptions 3.1, 3.2, 3.3, and 3.5 hold.

Consider the probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \Pr)$ where $(\mathcal{F}_t)_{t\geq 0}$ is the canonical filtration. Then F_{η^π} satisfies the following PDE,

$$(\mathscr{L}_X F_{\eta^\pi}(\cdot, z))(x) - (r(x) + z\log\gamma)\frac{\partial}{\partial z}F_{\eta^\pi}(x,z) = 0 \qquad \Pr\text{-}\textit{almost surely} \tag{3.6}$$

where $F_{\eta^\pi}(x,z) = \eta^\pi([V_{\min}, z] \mid x)$.[4]

To aid in the proof of this theorem, we'll first prove some lemmas.

Lemma 3.2. *Let $(J_t)_{t\geq 0} = (X_t, \overline{G}_t)_{t\geq 0}$ be the truncated return process defined in Theorem 3.2. Then $\left(\overline{G}_t\right)_{t\geq 0}$ is a finite variation process.*

Proof. By definition, we have

$$\overline{G}_t = \int_0^t \gamma^s r(X_s) ds$$

Consider the measurable space $(\mathbf{R}_+, \Sigma)$ where Σ is the σ-algebra of Lebesgue-measurable subsets of the nonnegative reals, and let Λ denote the Lebesgue measure. We will use

[4]Note that $F_{\eta^\pi}(x, \cdot)$ is the CDF of the random return at state x.

$(\mathbf{R}_+, \Sigma)$ to measure *time*. By the Radon-Nikodym theorem, for each sample path $\omega \in \Omega$ (see Theorem 3.2), the function $\mu_\omega : \Sigma \to \mathbf{R}$ shown below is a signed measure on this measurable space,

$$\mu_\omega(A) = \int_A \gamma^{s \wedge T(\omega)} r(X_{s \wedge T(\omega)}(\omega)) \Lambda(ds) \qquad A \in \Sigma$$

Then, for any $\omega \in \Omega$, the mapping $t \mapsto G_t(\omega) = \mu_\omega([0,t])$. This shows that each sample path is a function $a : t \mapsto \mu_\omega([0,t])$ for the measure μ_ω, so every sample path is a finite variation function by definition. □

Lemma 3.3. *The truncated return process $(J_t)_{t \geq 0}$ as defined in Theorem 3.2 is a Feller-Dynkin process.*

Proof. Consider the filtered probability space $\mathsf{P} = (\Omega, \mathcal{F}, (\mathcal{F}_t)_{t \geq 0}, \Pr)$ defined previously. Proposition 1 shows that $(J_t)_{t \geq 0}$ is a Markov process. It remains to show that it is a Feller-Dynkin process. First, we must show that its transition semigroup maps $(P_t)_{t \geq 0}$ are endomorphisms on $C_0(\mathcal{X} \times \mathcal{R})$. Let $\psi \in C_0(\mathcal{X} \times \mathcal{R})$.

Note that since $(X_t)_{t \geq 0}$ has continuous sample paths, $\left(\overline{G}_t\right)_{t \geq 0}$ has absolutely continuous sample paths since

$$G_t(\omega) = \int_0^t \gamma^s r(X_s(\omega)) ds \qquad \omega \in \Omega$$

so it is bounded by the integral of a bounded function. Therefore $P_\delta \psi$ can be expressed as

$$P_\delta \psi = \int \psi \circ (X_{t+\delta}, \overline{G}_{t+\delta}) d \Pr$$

Since the sample paths $X_{t+\delta}, \overline{G}_{t+\delta}$ are continuous, the integrand above is a continuous function. Additionally, since $\psi, \mathcal{X}, \mathcal{R}$ are all compactly supported, we see that $P_\delta \psi$ is as well. Therefore $P_\delta \psi \in C_0(\mathcal{X} \times \mathcal{R})$.

It is easy to check that $P_0 \psi = \mathsf{id}$. This follows simply from the fact that $(X_t)_{t \geq 0}$ is a Feller-Dynkin process (so its semigroup has an identity) and $\left(\overline{G}_t\right)_{t \geq 0}$ is deterministic given $(X_t)_{t \geq 0}$. For the same reason, it follows that $P_t P_s = P_{t+s}$.

It remains to show that $\| P_\delta \psi - P_0 \psi \| \xrightarrow{\delta \downarrow 0} 0$. We have

$$\begin{aligned}\|P_\delta\psi - P_0\psi\| &= \|P_\delta\psi - \psi\| \\ &= \left\| \int \left(\psi \circ (X_{t+\delta}, \overline{G}_{t+\delta}) - \psi(X_t, \overline{G}_t)\right) d\Pr \right\| \\ &= \left\| \int \psi \circ (X_{t+\delta}, \overline{G}_{t+\delta}) d\Pr - \psi(X_t, \overline{G}_t) \right\|\end{aligned}$$

Since ψ is supported on a compact finite-dimensional set and it is continuous, it follows that it is bounded. Therefore, it follows by the dominated convergence theorem that

$$\begin{aligned}\lim_{\delta\to 0} \int \psi \circ (X_{t+\delta}, \overline{G}_{t+\delta}) d\Pr &= \int \psi \circ \lim_{\delta\to 0} (X_{t+\delta}, \overline{G}_{t+\delta}) d\Pr \\ &= \int \psi(X_t, \overline{G}_t) d\Pr \\ &= \psi(X_t, \overline{G}_t)\end{aligned}$$

This proves the claim. □

Corollary 3.1. *The truncated return process $(J_t)_{t\geq 0}$ defined in Theorem 3.2 has an infinitesimal generator $\mathscr{L} : C_0(\mathcal{X}\times\mathcal{R}) \to C_0(\mathcal{X}\times\mathcal{R})$ given by*

$$\mathscr{L}\psi(x,\overline{g}) = (\mathscr{L}_X \psi(\cdot,\overline{g}))(x) + r(x)\frac{\partial}{\partial \overline{g}}\psi(x,\overline{g}) \tag{3.7}$$

where $\mathscr{L}_X$ is the infinitesimal generator of the process $(\iota_1 J_t)_{t\geq 0} = (X_t)_{t\geq 0}$.

Proof. Since Lemma 3.3 shows that $(J_t)_{t\geq 0}$ is a Feller-Dynkin process, the existence of an infinitesimal generator driving this process is guaranteed. Let $\psi \in C_0^2(\mathcal{X}\times\mathcal{R})$. Then

$$\begin{aligned}\frac{P_\delta\psi(j) - \psi(j)}{\delta} &= \frac{1}{\delta}\Big(\mathbf{E}\left[\psi(J_{t+\delta}) \mid J_t = j\right] - \psi(j)\Big) \\ &= \mathbf{E}\left[\frac{1}{\delta}\left(\psi(J_{t+\delta}) - \psi(J_t)\right) \,\middle|\, J_t = j\right] \qquad (*)\end{aligned}$$

We will proceed by applying Itô's Lemma to this expectation. However, we must first verify that $(J_t)_{t\geq 0}$ satisfies the hypotheses of Itô's Lemma, namely, it must be a semi-martingale. It is easy to verify that this is the case. We can express $(J_t)_{t\geq 0}$ as

$$J_t = \overbrace{(X_t - \mathbf{E}[X_t], 0)^\top}^{M_t} + \overbrace{(\mathbf{E}[X_t], \overline{G}_t)^\top}^{A_t}$$

It follows immediately from Lemma 3.2 that $(A_t)_{t\geq 0}$ is a finite variation process. Furthermore, since $(X_t)_{t\geq 0}$ is a Feller-Dynkin process, we know from Lemma 3.1 that $(X_t - \mathbf{E}[X_t])_{t\geq 0}$ is a martingale. Thus, $(J_t)_{t\geq 0}$ can be expressed as a sum of a local martingale[5] and a finite variation process, making it a semimartingale by definition.

Since $(J_t)_{t\geq 0}$ is a semimartingale and $\psi \in C_0^2(\mathcal{X}\times\mathcal{R})$, we may apply Itô's lemma to expand $(*)$ as follows,

$$\begin{aligned}
\frac{P_\delta\psi(j) - \psi(j)}{\delta} &= \frac{1}{\delta}\mathbf{E}\left[\int_t^{t+\delta}\sum_{i=1}^{d+1}\frac{\partial\psi(J_s)}{\partial j^i}dJ_s^i + \frac{1}{2}\int_t^{t+\delta}\sum_{i=1}^{d+1}\sum_{k=1}^{d+1}\frac{\partial^2\psi(J_s)}{\partial j^i\partial j^k}d[J^i,J^k]_s \,\middle|\, J_t = j\right] \\
&= \overbrace{\frac{1}{\delta}\mathbf{E}\left[\int_t^{t+\delta}\sum_{i=1}^{d}\frac{\partial\psi(J_s)}{\partial j^i}dJ_s^i + \frac{1}{2}\int_t^{t+\delta}\sum_{i=1}^{d}\sum_{k=1}^{d}\frac{\partial^2\psi(J_s)}{\partial j^i\partial j^k}d[J^i,J^k]_s \,\middle|\, J_t = j\right]}^{a} \\
&\quad + \overbrace{\frac{1}{\delta}\mathbf{E}\left[\int_t^{t+\delta}\frac{\partial\psi(J_s)}{\partial j^{d+1}}dJ_s^{d+1} + \frac{1}{2}\frac{\partial^2\psi(J_s)}{\partial (j^{d+1})^2}d[J^{d+1},J^{d+1}]_s \,\middle|\, J_t = j\right]}^{b} \\
&\quad + \overbrace{\frac{1}{2\delta}\mathbf{E}\left[\int_t^{t+\delta}\sum_{i=1}^{d}\frac{\partial^2\psi(J_s)}{\partial j^i\partial j^{d+1}}d[J^i,J^{d+1}]_s \,\middle|\, J_t = j\right]}^{c}
\end{aligned}$$

Recall that $J_t^{1:d} = \iota_1 J_t = X_t$, and $J_t^{d+1} = \iota_2 J_t = \overline{G}_t$. In the limit as $\delta \downarrow 0$, the term a above therefore is simply the generator of the process $(X_t)_{t\geq 0}$ applied to ψ. Moreover, since it was shown that $(\overline{G}_t)_{t\geq 0}$ is a finite variation process in Lemma 3.2, it follows that $[J^i, J^{d+1}] \equiv 0$ for any $i \in [d+1]$ [Le Gall, 2016]. Consequently, we have $c \equiv 0$. Simplifying,

[5] By the definition of a local martingale, given in Appendix C.1.2, it is clear that all martingales are local martingales.

$$\begin{aligned}\lim_{\delta\to 0}\frac{P_\delta\psi(j)-\psi(j)}{\delta} &= \mathscr{L}_X\psi(j)+\lim_{\delta\to 0}\frac{1}{\delta}\mathbf{E}\left[\int_t^{t+\delta}\frac{\partial\psi(J_s)}{\partial\overline{g}}d\overline{G}_s \;\middle|\; J_t=j\right]+\frac{\partial\psi(j)}{\partial t}\\ &= \mathscr{L}_X\psi(j)+\frac{\partial\psi(j)}{\partial\overline{g}}r(X_t)\end{aligned}$$

This completes the proof. □

Now we're ready to prove the main result of this section.

Proof of Theorem 3.2. We want to study the probability measure $\eta^\pi(\cdot \mid x)$, where x can be an arbitrary state in $\mathcal{X}$. Recall that the truncated return process is defined such that

$$G_x^\pi \overset{\mathcal{L}}{=} \overline{G}_T \;\middle|\; X_0 = x$$

It's important to note the condition that $X_0 = x$. In particular

$$G_{x_t}^\pi \overset{\mathcal{L}}{\neq} \overline{G}_T$$

Rather, we have, for $t \leq T$

$$\begin{aligned}\overline{G}_T &\overset{\mathcal{L}}{=} \int_0^T \gamma^s r(X_s)ds\\ &\overset{\mathcal{L}}{=} \int_0^t \gamma^s r(X_s)ds + \int_t^T \gamma^s r(X_s)ds\\ &\overset{\mathcal{L}}{=} \overline{G}_t + \gamma^t\int_0^{T-t}\gamma^s r(X_{s+t})ds\end{aligned}$$

Therefore, the *time-adjusted random return* is expressed by

$$\frac{\overline{G}_T - \overline{G}_t}{\gamma^t} \stackrel{\mathcal{L}}{=} G^\pi_{x_t} \;\Big|\; X_0 = x_t, \overline{G}_0 = 0$$

We'll express the return measure function as the density of the time-adjusted random return. For any Borel set $A \subset \mathcal{R}$, we have

$$\eta^\pi(A \mid j) = \mathbf{E}\left[\phi_z(\overline{G}_T) \;\middle|\; J_t = j\right]$$

$$\phi_z = \chi_{\gamma^{-t}(A - \overline{G}_t)}$$
$$\gamma^{-t}(A - \overline{G}_t) = \{\gamma^{-t}(z - \overline{G}_t) : z \in A\}$$

Note that F_{η^π} is a solution to the Kolmogorov backward equation for $(J_t)_{t\geq 0}$. However, we want to express η^π as the solution to an equation governed by the process $\gamma^{-t}(Z_t(z))_{t\geq 0}$ where $Z_t(z) = \gamma^{-t}(z - \overline{G}_t)$ for any return z. By applying the Feynman-Kac formula (shown in Theorem C.2) to the generator derived in Lemma 3.3, the generator $\mathscr{L}^\star$ corresponding to the process $(X_t, Z_t(z))$ is given by

$$\mathscr{L}^\star = \overline{\mathscr{L}} - \log \gamma \iota_2$$

where $\overline{\mathscr{L}}\psi(x, z) = \mathscr{L}\psi(x, -z)$ since $\frac{dz}{d\overline{g}} = -1$.

Finally, since $\eta^\pi(\cdot \mid x)$ is supposed to be a stationary distribution, the Kolmogorov backward equation for the generator $\mathscr{L}^\star$ becomes

$$\begin{aligned} 0 = \frac{\partial}{\partial t} F_{\eta^\pi}(x, z) &= -\mathscr{L}^\star_Z F_{\eta^\pi}(x, z) \\ &= \mathscr{L}_X F_{\eta^\pi}(x, z) - (r(x) + z \log \gamma) \frac{\partial}{\partial z} F_{\eta^\pi}(x, z), \end{aligned}$$

as claimed. Since $\eta^\pi(\cdot \mid x)$ is assumed to be absolutely continuous, the existence of $\frac{\partial F_{\eta^\pi}}{\partial z}$ is guaranteed. □

The process $(X_t, Z_t(z))_{t\geq 0}$ used in this proof will be referred to henceforth as the *conditional backward return process*. Following is its formal definition.

Definition 12 (Conditional Backward Return Process)**.** Let $(J_t)_{t\geq 0} = (X_t, \overline{G}_t)_{t\geq 0}$ denote the truncated return process with a discount factor γ induced by an agent following a fixed policy to produce the Markov process $(X_t)_{t\geq 0} \subset \mathcal{X}$. The *conditional backward return process* conditioned on the return taking value $z \in \mathcal{R}$ is the process $(\Upsilon(z)_t)_{t\geq 0} : \mathbf{R}_+ \to \mathcal{X} \times \mathcal{R}$ given by

$$\Upsilon(z)_t = (X_t, \gamma^{-t}(z - \overline{G}_t))$$

$\triangledown$

Unlike the truncated return process which accumulates rewards "forward in time", the conditional backward return process conditions on a given return z and describes the return left to be obtained in order to attain a return of z.

Corollary 3.2 (The Distributional HJB Equation for Itô Diffusions)**.** *Under the assumptions of Theorem 3.2 as well as Assumption 3.4, the stationary return distribution function η^π satisfies the following equation,*

$$0 = \langle \nabla_x F_{\eta^\pi}(x,z), f_\pi(x)\rangle + \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \mathsf{H}_x F_{\eta^\pi}(x,z)\boldsymbol{\sigma}_\pi(x)\right) - (r(x) + z\log\gamma)\frac{\partial}{\partial z}F_{\eta^\pi}(x,z) \tag{3.8}$$

Proof. This result follows directly from Theorem 3.2, since the infinitesimal generator $\mathscr{L}_X$ of an Itô Diffusion $(X_t)_{t\geq 0}$ governed by

$$dX_t = f_\pi(X_t)dt + \boldsymbol{\sigma}_\pi(X_t)dB_t$$

is known [Rogers and Williams, 1994, Villani, 2008, Jordan et al., 2002] to be

$$\mathscr{L}_X\phi = \langle \nabla\phi, f_\pi\rangle + \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi^\top \mathsf{H}\phi\boldsymbol{\sigma}_\pi\right)$$

$\square$

Remark 3.2. Readers that are familiar with optimal control theory may notice a similarity between (3.8) and the HJB equation [Fleming and Soner, 2006]. In fact, it can be seen that in the case of deterministic dynamics, (3.8) is equivalent to the deterministic HJB equation in the policy evaluation setting, in a weak sense. When the dynamics are deterministic, we have $\boldsymbol{\sigma}_\pi \equiv 0$, and the return distribution function is given by $\varrho^\pi(\cdot \mid x) = \frac{\partial}{\partial z} F_{\eta^\pi}(x, z) = \delta_{V^\pi(x)}$, where $V^\pi(x) = \int_0^T \gamma^s r(X_s) ds$ is the value function. When $z = V^\pi(x)$, (3.6) reduces to

$$\begin{aligned} 0 &= -\langle \nabla_x V^\pi(x), f_\pi(x) \rangle - r(x) - \log \gamma V^\pi(x) \\ &= \langle \nabla_x V^\pi(x), f_\pi(x) \rangle + r(x) + \log \gamma V^\pi(x) \end{aligned}$$

which is precisely the HJB equation with an infinite time horizon [Munos, 2004, Theorem 1] in the policy evaluation setting.

For $z \neq V^\pi(x)$, we are left with $\langle \nabla_x V^\pi(x), f_\pi(x) \rangle = 0$, which simply states that the agent is moving orthogonally to the direction of steepest ascent of the value function.

4

Approximate Distributional Dynamic Programming

In order to construct and analyze distributional reinforcement learning in continuous time, we may compute the return distribution function by solving (3.6) at each state. Many algorithms exist for solving PDEs, many examples can be readily found within the stochastic optimal control literature [Fleming and Soner, 2006]. An additional challenge in the distributional RL setting is that solutions to (3.6) belong to a constrained set – that being the set of probability measures over $\mathcal{R}$. The algorithm of Benamou and Brenier [Benamou and Brenier, 2000] addresses such constraints, however this algorithm works only for fixed, finite time intervals, and scales poorly with the episode length. In this chapter, we will construct a tractable method for approximating solutions to (3.6) via gradient-based iterative refinements, inspired by the results discussed in §2.5.

As in the case of discrete-time distributional reinforcement learning, it is impossible to learn a return distribution function exactly since the space of probability measures over $\mathcal{R}$ is infinite-dimensional. The continuous time dynamics introduces the additional challenge of time discretization. In order to proceed, we will have to resort to *approximately*

computing the return distribution function, and this chapter will discuss how this can be accomplished.

The remainder of this chapter will be structured as follows:

- We will begin by illustrating a framework for **representing probability measures** in §4.1, and we will discuss how **the choice of representation affects the characterization of return distribution evolution** in continuous time;
- A continuous-time formulation of **distributional policy evaluation** is analyzed in §4.2;
- A brief discussion of the **distributional optimal control problem** in §4.3 concludes the chapter.

4.1 Representation of Probability Measures

In order to represent probability measures in a tractable manner, we follow the framework suggested by Rowland et al. [2019] and explicitly distinguish between the statistics of a random variable and its distribution. In doing so, we restrict the class of probability measures that can be modeled to a class of probability measures that can be *imputed* from a finite set of *statistical functionals*.

Definition 13 (Statistical Functional)**.** Let Ω denote a measurable space. A *statistical functional* is a function $s : \mathcal{P}_p(\Omega) \to \mathbf{R}$. The values taken by statistical functionals are called *statistics*. $\triangledown$

Definition 14 (Imputation Strategy, Rowland et al. [2019])**.** Let Ω be a measurable space and $N \in \mathbf{N}$. An *imputation strategy* is a function $\Phi : \mathscr{S}_\Phi \to \mathcal{P}_p(\Omega)$ such that for any set of statistical functionals $\mathbf{s} = \{s_n\}_{n=1}^N$,

$$\mathbf{s} \circ \Phi = \mathsf{id}$$

where $\mathscr{S}_\Phi \subset \mathbf{R}^N$ is the set of *admissible statistics* corresponding to the imputation strategy Φ. For example, the imputation strategy $\Psi : (s_1, s_2) \mapsto \mathcal{N}(s_1, s_2)$ has the domain $\mathscr{S}_\Psi = \mathbf{R} \times (0, \infty)$, since the second parameter of Ψ corresponds to the variance of a normal distribution, which must be positive.

Simply put, an imputation strategy maps a set of statistics to a probability measure with those statistics. ▽

Imputation strategies have a very simple definition in the language of categories which provides a nice pictorial description.

Proposition 3 (A categorical perspective on Imputation Strategies)**.** *Let* SF *and* Pr *denote the category of sets of statistical functionals and the category of sets of probability measures respectively, where both are understood as subcategories of* Set*. An imputation strategy is a* ***functor*** $\Phi : \mathsf{SF} \to \mathsf{Pr}$*. That is,* Φ *is a function for which diagrams of the form shown in Figure 4.1 commute*[1]*.*

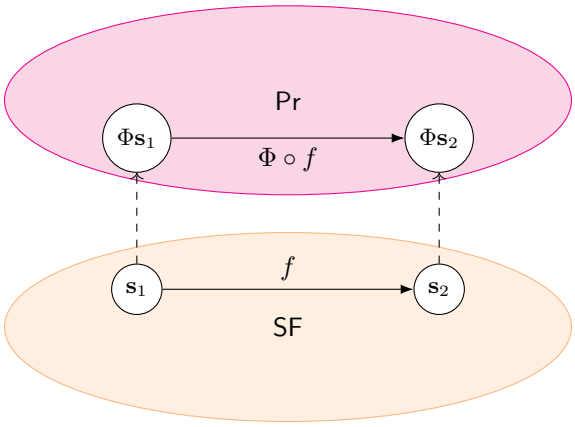


Figure 4.1: Imputation strategy as a functor

There is a number of natural choices for combinations of imputation strategies and statistical functionals, and any particular choice may have considerable theoretical or computational implications. A few of them are described below.

The mean A very simple choice for the set statistical functionals is the singleton containing the mean functional, $\mathbf{s} = \{\eta \mapsto \mathbf{E}_{Z\sim\eta}[Z]\}$. In fact, distributional RL with this representation is equivalent to standard RL.

A set of moments A natural extension from the representation of solely the mean is a representation consisting of a finite number of moments, $\mathbf{s}(\eta) = \{\mathbf{E}_{Z\sim\eta}[Z^{n_i}] : i \in [N]\}$ where $\{n_i\}_{i=1}^{N} \subset \mathbf{N}$. Imputation strategies for this representation are non-trivial.

A set of atoms A familiar finite-dimensional parameterization of a probability measure over an arbitrary measurable space $(\Omega, \mathcal{F})$ with a σ-finite measure μ has the form

[1] A diagram commutes if for any two nodes (objects) in the diagram, the composition of arrows on any path between those nodes is the same.

$$\hat{\eta}(\cdot) = \sum_{i=1}^{N} \alpha_i \chi_{p_i} \tag{4.1}$$

where $\{\alpha_i\}_{i=1}^N \subset \mathbf{R}_+$, $\sum_i \alpha_i = 1$, $\{p_i\}_{i=1}^N \subset \mathcal{F}$ is a partition[2] of Ω, and $\mu(p_i) = \mu(p_1)$ for each $i \in [N]$. For probability measures over bounded subsets of $\mathbf{R}$, this is equivalent to splitting the subset into intervals of equal length and modeling the probabily masses of a random variable taking a value in each interval. Mathematically, the corresponding statistical functionals are

$$s_i(\hat{\eta}) = \mathbf{E}_{Z \sim \hat{\eta}} \left[\chi_{p_i} \right]$$

and the imputation strategy is $\Phi(\mathbf{s}) = \sum_i s_i \chi_{p_i}$. An issue with this scheme is that until the return distribution function estimate has converged, operator applications (such as distributional Bellman operators) will yield distributions that are supported on a different set of atoms, which makes these distributions difficult to interpret with respect to the set of statistical functionals. Nonetheless, this approach was taken in the first approach to distributional RL, namely *Categorical Distributional RL* [Bellemare et al., 2017a] and the C51 algorithm, which solves the discrepancy of support problem by introducing a projection operator that maps categorical distributions to an appropriate set of statistics.

Quantiles A very simple imputation strategy can be leveraged if we model the statistical functionals corresponding to evenly spaced *quantiles* of a random variable. Let $Z \in A \subset \mathbf{R}$ be a random variable with $\text{Law}(Z) = \hat{\eta}$, where A is a compact set. The τ-quantile $q_{\hat{\eta}}(\tau_i)$ of $\hat{\eta}$ is defined as

$$q_{\hat{\eta}}(\tau) = \inf \{ z \in A : \hat{\eta}(Z \leq z) = \tau \}$$

It is also known from Koenker and Bassett Jr [1978], Dabney et al. [2018a] that quantiles can also be expressed via an optimization of the form

[2] A *partition* of a set A is a collection $\{A_i\}_{i=1}^N$ such that $i \neq j \implies A_i \cap A_j = \emptyset$ and $\bigcup_i A_i = A$.

$$q_{\hat{\eta}}(\tau) = \arg\min_{z \in A} \mathop{\mathbf{E}}_{Z \sim \hat{\eta}} \left[\left(\tau \mathbf{1}_{[Z > z]} + (1 - \tau) \mathbf{1}_{[Z \leq z]} \right) |Z - z| \right]$$

We can choose our statistical functionals $\{s_i\}_{i=1}^N$ such that $s_i(\hat{\eta}) = q_{\hat{\eta}}(\hat{\tau}_i)$, where $\tau_i = (i-1)/N$ for $i \in [N+1]$, and $\hat{\tau}_i = (\tau_{i+1} + \tau_i)/2$ for $i \in [N]$. Dabney et al. [2018a] shows that this choice of statistical functionals minimizes the 1-Wasserstein distance between a distribution and its approximation with a finite number of uniformly-weighted point masses.

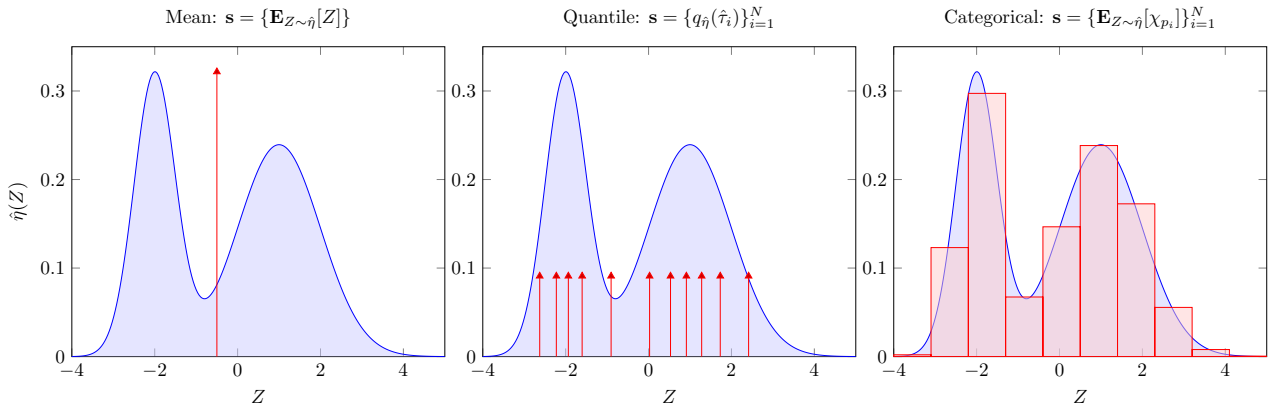


Figure 4.2: Examples of imputed probability measures

Perhaps the most immediate question is whether or not the statistical functionals can be learned exactly (that is, they converge to the statistics of the target distribution) by successive Bellman-like dynamic programming updates. This property is formalized by the following definition.

Definition 15 (Bellman-Closedness, Rowland et al. [2019])**.** A set of statistical functionals is said to be *Bellman-closed* if for any MDP and state x in the MDP the statistics $\mathbf{s}(\eta(\cdot \mid x))$ can be expressed exactly in terms of the discount factor γ, $\mathbf{s}(\eta(\cdot \mid X_1)) \mid X_0 = x$, and $R_0 = \int_0^1 \gamma^s r(X_s) ds$. ▽

Notably, there are remarkably few Bellman-closed sets of statistical functionals.

Theorem 4.1 (Rowland et al. [2019])**.** *Among all finite sets of statistical functionals* $\mathbf{s} = \{s_i\}_{i=1}^N$ *having the form* $\mathbf{s}(\eta) = \mathbf{E}_{Z\sim\eta}[h(Z)]$ *for some measurable function* h*,* $\mathbf{s}$ *is Bellman closed* only if *it has the same span as that of the first* N *moment functionals.*

This tells us that neither the quantile nor the categorical representations are Bellman-closed. While this is unfortunate, distributional RL algorithms using these represen-

tations tend to approximate the true return distributions quite well empirically [Hessel et al., 2018, Bellemare et al., 2020]. It turns out that these representations are *approximately* Bellman-closed, and Rowland et al. [2019] provides statistical rates based on the number of statistical functionals used in the representation and the discount factor.

4.1.1 Implications of the Representation in Continuous-Time

At first glance, the method we choose to represent return distributions may seem completely independent of the continuous-time problem. However, this is not the case, and we will see that under some representations the problem becomes much more complex.

Consider once again the distributional HJB equation (3.6). Fix a set of statistical functionals $\mathsf{s} = \{s_i\}_{i=1}^N$ and suppose now that $\eta(\cdot \mid x) = \Phi(\zeta(x))$ where Φ is an imputation strategy and $\zeta(x) : x \mapsto \mathsf{s}(\eta(\cdot \mid x))$. The learning problem reduces to learning ζ.

Searching over a space of admissible statistics can be a lot more convenient than searching over a space of probability measures. Indeed, many spaces of admissible statistics are Euclidean, which is far from the case for spaces of probability measures. Because of this, operations in the space of admissible statistics tend to be much more intuitive. As such, a characterisation of return distribution functions in terms of statistical functionals will be very useful for designing algorithms. Theorem 4.2 demonstrates such a characterization.

In Theorem 4.2, we make use of vectorized equations, which consist of notation that is common in multivariable calculus and linear algebra. In particular, recall that the *Jacobian* $\mathsf{J}\vec{v}$ of a vector-valued function $\vec{v} : \mathbf{R}^m \to \mathbf{R}^n$ is a matrix in $\mathbf{R}^{n \times m}$ such that $[\mathsf{J}\vec{v}(x)]_{ij} = \frac{\partial \vec{v}_i}{\partial x_j}(x)$, and the Hessian $\mathsf{H}f$ of a function $f : \mathbf{R}^m \to \mathbf{R}$ is a matrix in $\mathbf{R}^{m \times m}$ where $[\mathsf{H}f(x)]_{ij} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j}$. Moreover, in order to derive a robust characterization of the return distribution, we will require a mild regularity condition on the imputation strategy.

Definition 16 (Statistical Smoothness)**.** An imputation strategy $\Phi : \mathscr{S}_\Phi \to \mathcal{P}_p(\mathcal{R})$ is called *statistically smooth* if $\Phi(s)$ is a tempered distribution (see Appendix D) for each $s \in \mathscr{S}_\Phi$. Likewise, a return distribution function η is said to be statistically smooth if $F_{\eta^\pi}\eta(x, \cdot)$ is a tempered distribution for each $x \in \mathcal{X}$ and $F_{\eta^\pi}\eta(\cdot, z)$ is twice continuously differentiable almost everywhere for each $z \in \mathcal{R}$. ▽

Additionally, we will introduce the following terms for the purpose of improving legibility and garnering intuition.

Definition 17 (Spatial Diffusivity)**.** Let $\Phi : \mathscr{S}_\Phi \to \mathcal{P}_p(\mathcal{R})$ be a statistically smooth imputation strategy and $(X_t)_{t \geq 0} \subset \mathcal{X} \subset \mathbf{R}^d$ an Itô diffusion given by

$$dX_t = f_\pi(X_t)dt + \boldsymbol{\sigma}_\pi(X_t)dB_t$$

The *spatial diffusivity* of the random return under the imputation strategy Φ is defined as the mapping $\mathbf{K}^x_{\eta^\pi} : \mathcal{X} \times \mathcal{R} \to \mathbf{R}^{d\times d}$ given by

$$\mathbf{K}^x_{\eta^\pi}(x, z) = \sum_{i=1}^{N} \nabla_i \Phi(\zeta(x))(z) \mathsf{H}\zeta^i(x)$$

$\triangledown$

More intuitively, the spatial diffusivity is a term defined by the stochasticity of the return due to the stochasticity of the state process. We will also identify a similar term due to the stochasticity of the return due to the variability of the statistics as a result of the stochasticity in the state process.

Definition 18 (Statistical Diffusivity)**.** Let $\Phi : \mathscr{S}_\Phi \to \mathcal{P}_p(\mathcal{R})$ be a statistically smooth imputation strategy and $(X_t)_{t\geq 0} \subset \mathcal{X} \subset \mathbf{R}^d$ an Itô diffusion like that of Definition 17. The *statistical diffusivity* of the random return under the imputation strategy Φ is defined as the mapping $\mathbf{K}^s_{\eta^\pi} : \mathcal{X} \times \mathcal{R} \to \mathbf{R}^{d\times d}$ given by

$$\mathbf{K}^s_{\eta^\pi}(x, z) = \zeta_x(x)^\top \left(\frac{\partial^2}{\partial z^2} \Phi(\zeta(x))(z) \right) \zeta_x(x)$$

$\triangledown$

We can now analyze the return distribution functions characterized by (3.6) with respect to imputation strategies and statistical functionals.

Theorem 4.2 (The Statistical HJB Loss for Policy Evaluation)**.** *Let the assumptions of Corollary 3.2 hold. In particular, recall that the state dynamics of the agent following policy π are given by*

$$dX_t = f_\pi(X_t)dt + \boldsymbol{\sigma}_\pi(X_t)dB_t \qquad\qquad X_t \in \mathcal{X} \subset \mathbf{R}^d$$

For any statistically smooth imputation strategy $\Phi : \mathscr{S}_\Phi \to \mathcal{P}_p(\mathcal{R})$ on a space of admissible statistics $\mathscr{S}_\Phi \subset \mathbf{R}^N$ and a corresponding set of statistical functionals $\mathbf{s} : \mathcal{P}_p(\mathcal{R}) \to \mathscr{S}_\Phi$ as defined

above such that $\mathbf{s} \circ \Phi = \mathsf{id}$*, we define the following terms,*

$$\zeta(x) = \mathbf{s}(\eta^\pi(\cdot \mid x)) \qquad \mathcal{X} \to \mathscr{S}_\Phi \tag{4.2}$$

$$\zeta_x(x) = \mathsf{J}_x \zeta(x) \qquad \mathcal{X} \to \mathbf{R}^{N \times d} \tag{4.3}$$

We define the Statistical HJB Loss *by the following equation,*

$$\begin{aligned} \mathcal{L}_S(\zeta(x), \Phi) = \nabla_\zeta F_{\Phi(\zeta)}(x,z)^\top \zeta_x(x) f_\pi(x) - (r(x) + \log \gamma z) \otimes \frac{\partial}{\partial z} F_{\Phi(\zeta)}(x,z) \\ + \frac{1}{2} \boldsymbol{\sigma}_\pi(x)^\top \left(\mathbf{K}^x_{\eta^\pi}(x,z) + \mathbf{K}^s_{\eta^\pi}(x,z) \right) \boldsymbol{\sigma}_\pi(x) \end{aligned} \tag{4.4}$$

where $\mathbf{K}^x_{\eta^\pi}, \mathbf{K}^s_{\eta^\pi}$ *are the spatial diffusivity and the statistical diffusivity respectively.*

Then if F_η *satisfies* (3.6) *and* $F_{\eta(x)} = \Phi(\zeta(x))$*, it is necessary that* $\mathcal{L}_S(\zeta(x), \Phi) = 0$*.*

Proof. This is simply proved by applications of the chain rule to (3.6). □

Theorem 4.2 presents a condition for the return distribution function that is formulated as a loss, as opposed to a PDE, since generally the probability measures that we impute from a finite collection of statistics do not form a rich enough class to satisfy the distributional HJB equation exactly. Thus, we cannot expect to characterize these probability measures like we did in Theorem 3.2. Equation (4.4) can reasonably be interpreted as a loss function for distributional policy evaluation, since it is minimized when the statistics are sufficient to encode the return distribution function accurately.

It looks like we have taken a step backward here, as (4.4) appears substantially more complex than (3.6). However, it turns out that a weaker form of (4.4) exists that is greatly simplified when the imputation strategy has a particular structure.

Corollary 4.1. *In the context of Theorem 4.2, if* Φ *has the form*

$$\Phi(\zeta(x)) = \frac{1}{N} \sum_{i=1}^N \delta_{\zeta^i(x)}, \tag{4.5}$$

then at each state $x \in \mathcal{X}$*, the following system is satisfied by the statistics* $\{\zeta^i(x)\}_{i=1}^N$*:*

$$\begin{cases} 0 = \langle \nabla_i \zeta(x), f_\pi(x) \rangle + r(x) + \zeta^i(x) \log \gamma + \frac{1}{2} \operatorname{Tr} \left(\boldsymbol{\sigma}_\pi(x)^\top \mathsf{H}_x \zeta^i(x) \boldsymbol{\sigma}_\pi(x) \right) \\ \mathbf{s}^i(\eta(\cdot \mid x)) = \zeta^i(x) \\ i = 1, \dots, N \end{cases} \tag{4.6}$$

in the distributional sense (see Appendix D, definition 48).

Proof. Let $\phi : \mathcal{X} \times \mathcal{R} \to \mathbf{R}$ be an arbitrary test function in the Schwartz class $\mathscr{S}$, and let $\eta = \Phi(\zeta(x))$ such that F_η is a distributional solution to (3.8). Denote by $\vartheta : \mathbf{R} \to [0, 1]$ the Heaviside step function $\vartheta(z) = \mathbf{1}_{[z>0]}$. Then, we have that

$$\begin{aligned} 0 = \int_{\mathcal{X}\times\mathcal{R}} \Bigg[& \phi(x,z) \left\langle \nabla_x \sum_{k=1}^N \vartheta(z - \iota_k \zeta(x)), f_\pi(x) \right\rangle - \phi(x,z)(r(x) + z \log \gamma) \frac{\partial}{\partial z} \sum_{k=1}^N \vartheta(z - \iota_k \zeta(x)) \\ & + \frac{1}{2} \phi(x,z) \operatorname{Tr} \left(\boldsymbol{\sigma}_\pi(x)^\top \left(\mathsf{H}_x \sum_{k=1}^N \vartheta(z - \iota_k \zeta(x)) \right) \boldsymbol{\sigma}_\pi(x) \right) \Bigg] dz dx \\ = \int_{\mathcal{X}\times\mathcal{R}} \Bigg[& \left\langle \phi(x,z) \nabla_x \sum_{k=1}^N \vartheta(z - \iota_k \zeta(x)), f_\pi(x) \right\rangle - \phi(x,z)(r(x) + z \log \gamma) \frac{\partial}{\partial z} \sum_{k=1}^N \vartheta(z - \iota_k \zeta(x)) \\ & + \frac{1}{2} \operatorname{Tr} \left(\boldsymbol{\sigma}_\pi(x)^\top \phi(x,z) \left(\mathsf{H}_x \sum_{k=1}^N \vartheta(z - \iota_k \zeta(x)) \right) \boldsymbol{\sigma}_\pi(x) \right) \Bigg] dz dx \end{aligned}$$

Taking distributional derivatives once, the Heaviside step functions are transformed into Dirac distributions, yielding

$$\begin{aligned} 0 = \int_{\mathcal{X}\times\mathcal{R}} \Bigg[& \left\langle -\phi(x,z) \sum_{k=1}^N \delta_{\iota_k \zeta(x)}(z) \nabla_x \iota_k \zeta(x), f_\pi(x) \right\rangle - \phi(x,z)(r(x) + z \log \gamma) \sum_{k=1}^N \delta_{\iota_k \zeta(x)}(z) \\ & - \frac{1}{2} \operatorname{Tr} \left(\boldsymbol{\sigma}_\pi(x)^\top \phi(x,z) \left(\nabla_x \sum_{k=1}^N \delta_{\iota_k \zeta(x)}(z) \right) \nabla_x \iota_k \zeta(x) \boldsymbol{\sigma}_\pi(x) \right) \Bigg] dz dx \end{aligned}$$

Next, we carry out the second spatial derivative.

$$
\begin{aligned}
0 &= \int_{\mathcal{X}\times\mathcal{R}} \Bigg[\left\langle -\phi(x,z) \sum_{k=1}^{N} \delta_{\iota_k\zeta(x)}(z) \nabla_x \iota_k \zeta(x), f_\pi(x) \right\rangle - \phi(x,z)(r(x) + z\log\gamma) \sum_{k=1}^{N} \delta_{\iota_k\zeta(x)}(z) \\
&\qquad - \frac{1}{2} \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \phi(x,z) \left(\nabla_x \sum_{k=1}^{N} \delta_{\iota_k\zeta(x)}(z) \nabla_x \iota_k \zeta(x) \right) \boldsymbol{\sigma}_\pi(x) \right) \Bigg] dz dx \\
&= \int_{\mathcal{X}\times\mathcal{R}} \Bigg[\left\langle -\phi(x,z) \sum_{k=1}^{N} \delta_{\iota_k\zeta(x)}(z) \nabla_x \iota_k \zeta(x), f_\pi(x) \right\rangle - \phi(x,z)(r(x) + z\log\gamma) \sum_{k=1}^{N} \delta_{\iota_k\zeta(x)}(z) \\
&\qquad - \frac{1}{2} \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \phi(x,z) \sum_{k=1}^{N} \left[\nabla_x \delta_{\iota_k\zeta(x)}(z) \nabla_x \iota_k \zeta(x) + \delta_{\iota_k\zeta(x)}(z) \mathsf{H}_x \iota_k \zeta(x) \right] \boldsymbol{\sigma}_\pi(x) \right) \Bigg] dz dx \\
&= \int_{\mathcal{X}\times\mathcal{R}} \phi(x,z) \Bigg[\left\langle \sum_{k=1}^{N} \delta_{\iota_k\zeta(x)}(z) \nabla_x \iota_k \zeta(x), f_\pi(x) \right\rangle + (r(x) + z\log\gamma) \sum_{k=1}^{N} \delta_{\iota_k\zeta(x)}(z) \\
&\qquad + \frac{1}{2} \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \sum_{k=1}^{N} \delta_{\iota_k\zeta(x)}(z) \mathsf{H}_x \iota_k \zeta(x) \boldsymbol{\sigma}_\pi(x) \right) \Bigg] dz dx \\
&\qquad + \frac{1}{2} \overbrace{\int_{\mathcal{X}\times\mathcal{R}} \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \phi(x,z) \nabla_x \delta_{\iota_k\zeta(x)}(z) \nabla_x \zeta(x) \boldsymbol{\sigma}_\pi(x) \right) dz dx}^{(a)}
\end{aligned}
$$

We isolate the term (a) as it involves the (distributional) derivative of the Dirac distribution, which is a strange object. However, since our equation holds for any test function ϕ, we will show that, with the right choice of test function, $(a) = 0$.

Choose any $\overline{x} \in \mathcal{X}$ and let $\epsilon > 0$. Then let $\phi(x,z) = \varrho_\epsilon(x)\psi(z)$ where $\varrho_\epsilon : \mathcal{X} \to \mathbf{R}$ and $\psi : \mathcal{R} \to \mathbf{R}$ are members of the Schwartz class $\mathscr{S}$. We define $\varrho_\epsilon(x)$ as follows,

$$
\varrho_\epsilon(x) = \frac{1}{\epsilon\sqrt{\pi}} \exp\left(-\frac{\|x - \overline{x}\|^2}{\epsilon^2} \right)
$$

It is well known that ϱ_ϵ is a Schwartz function [Lax and Sons, 2002]. Moreover, since $\nabla_x \varrho_\epsilon(\overline{x}) = 0$ and ϱ_ϵ is smooth, we can find a neighborhood B of $\overline{x}$ so small that $\sup_{x_1,x_2\in B} \|x_1 - x_2\| \leq \epsilon$. We are left with

$$
\begin{aligned}
(a) = \lim_{\epsilon \to 0} \Bigg[& \int_B \int_{\mathcal{R}} \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \phi(x,z) \nabla_x \delta_{\iota_k \zeta(x)}(z) \nabla_x \iota_k \zeta(x) \boldsymbol{\sigma}_\pi(x) \right) dz dx \\
& + \int_{\mathcal{X} \setminus B} \int_{\mathcal{R}} \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \phi(x,z) \nabla_x \delta_{\iota_k \zeta(x)}(z) \nabla_x \iota_k \zeta(x) \boldsymbol{\sigma}_\pi(x) \right) dz dx \Bigg] \\
= \lim_{\epsilon \to 0} \Bigg[& - \overbrace{\int_B \int_{\mathcal{R}} \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \psi(z) \nabla_x \varrho_{\epsilon(x)} \delta_{\iota_k \zeta(x)}(z) \nabla_x \iota_k \zeta(x) \boldsymbol{\sigma}_\pi(x) \right) dz dx}^{\mathcal{M}_\epsilon} \\
& - \overbrace{\int_{\mathcal{X} \setminus B} \int_{\mathcal{R}} \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \psi(z) \nabla_x \varrho_\epsilon(x) \delta_{\iota_k \zeta(x)}(z) \nabla_x \iota_k \zeta(x) \boldsymbol{\sigma}_\pi(x) \right) dz dx}^{\mathcal{E}_\epsilon} \Bigg]
\end{aligned}
$$

It is also well-known $\lim_{\epsilon \to 0} \varrho_\epsilon = \delta_{\overline{x}}$ [Lax and Sons, 2002]. Since necessarily $\overline{x} \notin \mathcal{X} \setminus B$, the term $\mathcal{E}_\epsilon$ vanishes. Given that $\sup_{x_1, x_2 \in B} \|x_1 - x_2\| \leq \epsilon$, we have

$$
\begin{aligned}
|\mathcal{M}_\epsilon| &\leq \epsilon \sup_{x \in B} \left| \int_{\mathcal{R}} \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \psi(z) \delta_{\iota_k \zeta(x)}(z) \nabla_x \iota_k \zeta(x) \boldsymbol{\sigma}_\pi(x) \right) dz \right| \\
&= \epsilon \sup_{x \in B} \left| \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \psi(\iota_k \zeta(x)) \nabla_x \iota_k \zeta(x) \boldsymbol{\sigma}_\pi(x) \right) dz \right|
\end{aligned}
$$

By the assumption that $\zeta(x)$ is almost-everywhere differentiable, the supremum above is bounded for almost every $\overline{x}$, and it follows that $|\mathcal{M}_\epsilon| \to 0$ almost surely.

We are left with the following equation:

$$
0 = \lim_{\epsilon \to 0} \int_{\mathcal{X}} \int_{\mathcal{R}} \varrho_\epsilon(x) \psi(z) \sum_{k=1}^{N} \delta_{\iota_k \zeta(x)}(z) \left[\langle \nabla_x \iota_k \zeta(x), f_\pi(x) \rangle + r(x) + z \log \gamma + \frac{1}{2} \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(x)^\top \mathsf{H}_x \iota_k \zeta(x) \boldsymbol{\sigma}_\pi(x) \right) \right] dz dx
$$

Given that $\Phi(\zeta(x))$ is statistically smooth, it is a tempered distribution, so this limit exists. We mentioned previously that $\varrho_\epsilon \to \delta_{\overline{x}}$, so we have

$$
0 = \int_{\mathcal{R}} \psi(z) \sum_{k=1}^{N} \delta_{\iota_k \zeta(\overline{x})}(z) \left[\langle \nabla_x \iota_k \zeta(\overline{x}), f_\pi(\overline{x}) \rangle + r(\overline{x}) + z \log \gamma + \frac{1}{2} \mathsf{Tr}\left(\boldsymbol{\sigma}_\pi(\overline{x})^\top \mathsf{H}_x \iota_k \zeta(\overline{x}) \boldsymbol{\sigma}_\pi(\overline{x}) \right) \right] dz
$$

It follows by definition that $\Phi(\zeta(x))$ is a distributional solution to

$$0 = \sum_{k=1}^{N} \delta_{\iota_k \zeta(\overline{x})}(z) \left[\langle \nabla_x \iota_k \zeta(\overline{x}), f_\pi(\overline{x}) \rangle + r(\overline{x}) + z \log \gamma + \frac{1}{2} \mathsf{Tr} \left(\boldsymbol{\sigma}_\pi(\overline{x})^\top \mathsf{H}_x \iota_k \zeta(\overline{x}) \boldsymbol{\sigma}_\pi(\overline{x}) \right) \right]$$

Note that the equation above is a sum of weighted Diracs. Thus, the only way for it to be satisfied is if each of the terms in the sum individually vanishes. So, we have shown that for each $k \in [N]$ and almost every $x \in \mathcal{X}$, the statistics function $\iota_k \zeta$ is a distributional solution of

$$0 = \langle \nabla_x \iota_k \zeta(x), f_\pi(x) \rangle + r(x) + \iota_k \zeta(x) \log \gamma + \frac{1}{2} \mathsf{Tr} \left(\boldsymbol{\sigma}_\pi(x)^\top \mathsf{H}_x \iota_k \zeta(x) \boldsymbol{\sigma}_\pi(x) \right)$$

This completes the proof. □

Remark 4.1. A similar statement to Corollary 4.1 *cannot* be made for arbitrary representations, even if they're finite-dimensional (such as a categorical distribution). The simplicity of (4.6) is a consequence of the statistical diffusivity of the return vanishing under the representation (4.5). Additionally, this representation admits a simplified form of the spatial diffusivity of the return.

4.2 Policy Evaluation

In this section, we will look at how policy evaluation can be achieved via the analysis of an optimization problem in the space of probability measures. As foreshadowed in §2.5, we will proceed by studying continuous-time distributional policy evaluation as a gradient flow in the space of probability measures. While discrete-time iterative RL algorithms estimate the return distribution function with a sequence of iterates $\{\eta_k\}_{k=1}^{\infty}$ where $\eta_{k+1} = \mathscr{T}^\pi \eta_k$, our continuous-time formulation of policy evaluation will take the form of a *curve* $(\eta_\tau)_{\tau \geq 0}$ in the space of probability measures satisfying the continuity equation akin to (CE),

$$\frac{\partial}{\partial \tau} \varrho_\tau(\cdot \mid x) + \nabla \cdot (\varrho_\tau(\cdot \mid x) \mathbf{v}(x)) = 0 \qquad \forall x \in \mathcal{X}$$

Here $\varrho_\tau(\cdot \mid x)$ corresponds to the density function of $\eta_\tau(\cdot \mid x)$, and $\mathbf{v}$ is a vector field that can be interpreted as an update rule for a collection of "particles" whose density is $\varrho_\tau(\cdot \mid x)$ [Santambrogio, 2015]. It is well-known that solutions to continuity equations are measure-preserving [Ullrich, 2011], which ensures that each point on the curve $(\eta_\tau)_{\tau \geq 0}$ is a proper probability measure.

We will consider once again the truncated return process $(J_t)_{t \geq 0} = (X_t, \overline{G}_t)_{t \geq 0}$ and let $\eta^\pi(A \mid x) = \Pr(\overline{G}_T \in A \mid X_0 = x)$ be the return measure function, where T is the stopping time defined in Proposition 2. As usual, we assume that $\eta^\pi(\cdot \mid x)$ is absolutely continuous with respect to the Lebesgue measure for each state $x \in \mathcal{X}$.

Our goal ultimately is to construct a process $(\eta_\tau)_{\tau \geq 0}$ such that $\eta_\tau \to \eta^\pi$ in a "reasonable" topology. This process should be understood as an analogue to the sequence $\{(\mathscr{T}^\pi)^k Q_0\}_{k=0}^\infty$ in discrete-time reinforcement learning. We are faced with the following challenges,

1. We must find a suitable topology in which convergence will hold in a meaningful sense (for instance, the topology induced by the total variation distance will not suffice, as explained in §2.4);
2. In order to produce a realizable algorithm for policy evaluation, we have to find a discrete-time approximation $\{\eta_k^\delta\}_{k=1}^\infty$ of $(\eta_\tau)_{\tau \geq 0}$ that converges to $(\eta_\tau)_{\tau \geq 0}$ in the limit of infinitesimal timesteps, where $\delta \to 0$.

Fortunately, following the results discussed in §2.5, we will simultaneously circumvent both issues by exhibiting a process $(\eta_\tau)_{\tau \geq 0}$ that is a *gradient flow* of a functional in the 2-Wasserstein space [Ambrosio et al., 2008]. In particular, as long as the functional is λ convex in the sense of definition 2.33 for some $\lambda > 0$, we can be assured that $\eta_\tau \to \eta^\pi$ in the 2-Wasserstein metric [Jordan et al., 2002, Santambrogio, 2015], and the generalized minizing movements of the JKO scheme will provide us with a convergent time-discretized approximation of $(\eta_\tau)_{\tau \geq 0}$.

Let $(\mathscr{T}_\tau^\pi)_{\tau \geq 0}$ denote the transition semigroup of the conditional backward return process $(\Upsilon(z)_t)_{t \geq 0}$. By definition, we have

$$\mathscr{T}_\tau^\pi \iota_2(x, z) = \mathbf{E}\left[\iota_2(X_\tau, Z_\tau) \mid X_0 = x, Z_0 = z\right]$$

Let $\zeta : x \mapsto \mathsf{s}(\eta(\cdot \mid x)) \in \mathbf{R}^N$ be a set of statistical functionals such that there exists

$\Phi : \mathbf{R}^N \to \mathcal{P}_p(\mathcal{R})$ such that $\mathbf{s} \circ \Phi = \mathsf{id}$ and $\Phi(\mathbf{s}) = \frac{1}{N}\sum_{i=1}^N \delta_{\mathbf{s}^i}$. Then we can aim to learn the return measure by ensuring that $|\frac{\partial}{\partial \tau}\eta_\tau| \to 0$. We can attempt this by considering the gradient flow of the following functional,

$$\mathscr{F}(\mu_\tau) = \int_{\mathcal{R}} \left| \frac{\partial}{\partial \tau} F_{\mu_\tau}(x, z) \right|^2 d\mu_\tau(z) \tag{4.7}$$

Note that the loss functional $\mathscr{F}$ of (4.7) is minimized when $\frac{\partial}{\partial \tau} F_{\mu_\tau}(x, \cdot) = 0$, which will correspond to the return distribution function estimates converging to stationary points. These stationary points may not necessarily be "optimal" in a meaningful sense, since $\mathscr{F}$ may have local minima. Moving forward, we will demonstrate a trajectory $(\eta_\tau)_{\tau \geq 0}$ that does indeed have a unique fixed point. In order to accomplish this, we must impose the characterization of return distribution functions developed in §3.2 and §4.1 on the continuity equation.

Theorem 4.3. *Let $(\mu_\tau)_{\tau\geq 0}, (\nu_\tau)_{\tau \geq 0} : \mathbf{R}_+ \to \mathbf{W}_2$ be curves in the space of probability measures. Suppose*

$$\frac{\partial}{\partial \tau} F_{\mu_\tau}(x, z) = \mathscr{L}_X F_{\mu_\tau}(x, z) - (r(x) + z \log \gamma) \frac{\partial}{\partial z} F_{\mu_\tau}(x, z)$$
$$\frac{\partial}{\partial \tau} F_{\nu_\tau}(x, z) = \mathscr{L}_X F_{\nu_\tau}(x, z) - (r(x) + z \log \gamma) \frac{\partial}{\partial z} F_{\nu_\tau}(x, z)$$

Then $\lim_{\tau \to \infty} d_{\mathbf{W}_2}(\mu_\tau, \nu_\tau) = 0$, and the distance decays exponentially. Moreover, $\lim_{\tau\to\infty} \mu_\tau = \lim_{\tau \to \infty} \nu_\tau$ exists and the limit is unique.

This proof will proceed in a few steps:

1. We will show that $\mathscr{F}$ as defined in (4.9) is λ-convex for a $\lambda > 0$;
2. Then we can establish that $(F_{\mu_\tau})_{\tau \geq 0}$ and $(F_{\nu_\tau})_{\tau \geq 0}$ are gradient flows, in an EVI sense, to a loss functional that approximates $\mathscr{F}$;
3. Finally, using the properties of EVI gradient flows, we will deduce that the 2-Wasserstein distance between μ_τ, ν_τ decays exponentially in time.

In order to prove 1, we will begin with the following lemma.

Lemma 4.1. *The function $\mathscr{L}^\star|_{\mathbf{W}_2} : \mathbf{W}_2 \to \mathcal{R}$ defined as*

$$\mathscr{L}^\star|_{\mathbf{w}_2}\,\eta = \mathscr{L}_X F_\eta - (r \circ \iota_1 + \log\gamma\iota_2)\frac{\partial F_\eta}{\partial z} \tag{4.8}$$

is convex.

Proof. To begin, note that

$$\begin{aligned}
\mathscr{T}_\delta^\pi\,\eta(z \mid x) &= \mathbf{E}\left[\eta(\gamma^{-\delta}(z - \overline{G}_\delta) \mid X_\delta) \;\middle|\; X_0 = x\right] \\
&= \int_{\mathcal{X}}\int_{\mathcal{R}} \eta(\gamma^{-\delta}(z-\overline{g}) \mid x')\Pr(x', \overline{g} \mid x) \\
&= \int_{\mathcal{X}}\int_{\mathcal{R}} {f_\delta^{\overline{g},\gamma}}_\sharp\,\eta(\cdot \mid x')d\Pr(x', \overline{g} \mid x)
\end{aligned}$$

where $f_\sharp\mu = \mu \circ f^{-1}$ is the *pushforward measure* of μ through f, and $f_\delta^{\overline{g},\gamma}$ is a continuous time extension of the function $f^{r,\gamma}$ introduced in Rowland et al. [2019] defined as

$$f_\delta^{\overline{g},\gamma}(z) = \overline{g} + \gamma^\delta z$$

Then it follows that

$$\frac{\mathscr{T}_\delta^\pi\,\eta(z \mid x) - \eta(z \mid x)}{\delta} = \frac{1}{\delta}\int_{\mathcal{X}}\int_{\mathcal{R}} \left({f_\delta^{\overline{g},\gamma}}_\sharp\,\eta(\cdot \mid x') - \eta(\cdot \mid x)\right) d\Pr(x', \overline{g} \mid x)$$

On the right side of the equation above, we see that for any $\delta > 0$ the mapping

$$\mathsf{B}_\delta : \eta(z \mid x) \mapsto \frac{1}{\delta}(\mathscr{T}_\delta^\pi\,\eta(z \mid x) - \eta(z \mid x))$$

is convex. We also know that this mapping converges to $\mathscr{L}^\star|_{\mathbf{w}_2}$ in measure from Theorem 3.2. Thus, for any $\delta > 0$, $\lambda \in [0, 1]$, and return measure functions η_1, η_2,

$$\begin{aligned}
\mathsf{B}_\delta(\lambda\,\eta_1(\cdot \mid x) + (1-\lambda)\,\eta_2(\cdot \mid x)) &\leq \lambda \mathsf{B}_\delta(\eta_1(\cdot \mid x)) + (1-\lambda)\mathsf{B}_\delta\,\eta_2(\cdot \mid x) \\
\therefore \lim_{\delta\to 0} \mathsf{B}_\delta(\lambda\,\eta_1(\cdot \mid x) + (1-\lambda)\,\eta_2(\cdot \mid x)) &\leq \lim_{\delta\to 0} \lambda \mathsf{B}_\delta(\eta_1(\cdot \mid x)) + \lim_{\delta\to 0}(1-\lambda)\mathsf{B}_\delta\,\eta_2(\cdot \mid x) \\
\therefore \mathscr{L}^\star|_{\mathbf{W}_2}(\lambda\,\eta_1(\cdot \mid x) + (1-\lambda)\,\eta_2(\cdot \mid x)) &\leq \lambda \mathscr{L}^\star|_{\mathbf{W}_2}\,\eta_1(\cdot \mid x) + (1-\lambda)\mathscr{L}^\star|_{\mathbf{W}_2}\,\eta_2(\cdot \mid x)
\end{aligned}$$

This shows that $\mathscr{L}^\star|_{\mathbf{W}_2}$ is indeed convex. □

The purpose of Lemma 4.1 is to facilitate the proof that $\mathscr{F}$ is a convex functional, which will be shown in the sequel.

Proof of Theorem 4.3. Consider the functional $\mathscr{F}_\beta : \mathbf{W}_2 \to \mathbf{R}_+$ defined as

$$\mathscr{F}_\beta(\eta) = \int_{\mathcal{R}} \frac{1}{2} \left(\overbrace{\mathscr{L}^\star|_{\mathbf{W}_2}\,\eta(z \mid x)}^{\phi(z)} \right)^2 \eta(dz \mid x) + \frac{1}{\beta} \overbrace{\int_{\mathcal{R}} -\eta(z \mid x) \log \eta(z \mid x) dz}^{\mathcal{H}(\eta(\cdot|x))} \tag{4.9}$$

This functional is known to correspond to an *entropy-regularized* optimal transport cost [Cuturi, 2013], where the cost function on the underlying space (the space of returns) is determined by ϕ. Additionally, (4.9) is known [Jordan et al., 2002] to be the formulation of the Fokker-Planck equation (FP) expressed as an EVI gradient flow in $\mathbf{W}_2$ whenever ϕ^2 is convex Santambrogio [2015].

By Lemma 4.1 we know that ϕ is convex and non-constant. Moreover, the square function $F : x \mapsto x^2$ satisfies

$$\begin{aligned}
x^2 + y^2 &= (x-y)^2 + 2xy \\
F(y) &= -F(x) + (x-y)^2 + \nabla F(x) y \\
&= F(x) + (x-y)^2 - 2x^2 + \nabla F(x) y \\
&= F(x) + \frac{\lambda}{2}(x-y)^2 + \nabla F(x)(y-x) \qquad\qquad \lambda = 2
\end{aligned}$$

So the square function is λ-convex for $\lambda = 2$, and therefore there exists $\lambda > 0$ such that ϕ^2 is λ-convex, completing step 1 of the proof.

Since the entropy functional $\mathcal{H}(\eta(\cdot \mid x))$ is also famously convex, the loss functional $\mathscr{F}_\beta$ is λ-convex. Consequently, (4.9) satisfies the EVI$_\lambda$ condition (see §2.5.1), so it corresponds

to an EVI gradient flow due to step 1. This completes the proof of step 2.

By the contraction property of EVI gradient flows shown in Theorem 2, we have

$$\frac{d}{d\tau} d^2_{\mathbf{W}_2}(\mu_\tau, \nu_\tau) \leq -4\lambda d^2_{\mathbf{W}_2}(\mu_\tau, \nu_\tau)$$

Moreover, we see that $d^2_{\mathbf{W}_2}(\mu_\tau, \nu_\tau) \leq \exp(-4\lambda d^2_{\mathbf{W}_2}(\mu_0, \nu_0))$. Since $\lambda > 0$, we affirm that $d_{\mathbf{W}_2}(\mu_t, \nu_t) \to 0$ at an exponential rate when $\frac{\partial}{\partial t}\mu_\tau = -\nabla \mathscr{F}_\beta(\mu_\tau)$ and $\frac{\partial}{\partial t}\nu_\tau = -\nabla \mathscr{F}_\beta(\nu_\tau)$ both in the EVI sense. The work of Cuturi [2013] shows that the curves $(\mu_\tau)_{\tau \geq 0}, (\nu_\tau)_{\tau \geq 0}$ converge to gradient flows of $\mathscr{F}$ when $\beta \to \infty$, so $d_{\mathbf{W}_2}(\mu_\tau, \nu_\tau) \to 0$ exponentially when $F_{\mu_\tau}, F_{\nu_\tau}$ satisfy the equations stated in the theorem (note that $\mathscr{F}_\beta \overset{\beta \uparrow \infty}{\longrightarrow} \mathscr{F}$ pointwise).

Finally, uniqueness of the gradient flow is confirmed by Grönwall's Lemma [Gronwall, 1919]. Since $\mathscr{F}_\beta$ is clearly minimized when $\phi \equiv 0$, it follows that the gradient flow converges to the return measure satisfying (3.6), which is η^π. □

To summarize, we have shown that the distributional HJB equation (3.6) can be solved by formulating it as the gradient flow of the limit of functionals $\mathscr{F}_\beta$ (4.9) as $\beta \to \infty$. Therefore, we can approximate solutions to the distributional HJB equation by solving a Fokker-Planck equation (FP) with variance parameter tending to 0. The evolution of $(\eta_\tau)_{\tau \geq 0}$ according to the EVI gradient flow is understood as the continuous-time analogue to policy evaluation, and we showed that the stationary point of this curve is η^π as intended. However, we have not yet described a method of approximate policy evaluation that is realizable on a computer, as this description of policy evaluation requires the evolution of η_τ to be a continuous-time curve – in other words, we must compute updates to η_τ continuously in time. In the following section, we will bootstrap the results from the analysis of the JKO scheme (see §2.5.2) to derive a time-discretized approximation to policy evaluation that converges to $(\eta_\tau)_{\tau \geq 0}$ as the time discretization parameter shrinks to zero.

4.2.1 Time Discretization

While Theorem 4.3 is promising, we are still assuming our return measure estimates can evolve continuously in time – this of course cannot be the case for any imaginable algorithm. It may be tempting to apply a gradient-descent-like algorithm to minimize $\mathscr{F}_\beta$, however this can be too crude – after all, the space of return measures is highly non-

Euclidean, so updating a return measure with a Euclidean gradient step likely will not result in another return measure.

Thankfully, $\mathbf{W}_2$ is a convex set, and therefore we can consider applying a *proximal* gradient optimization routine [Rockafellar, 2015]. In particular, such proximal gradient algorithms have been studied specifically in $\mathbf{W}_2$ [Santambrogio, 2016, Salim et al., 2020], *and* for the optimization of a generalization of $\mathscr{F}_\beta$ [Jordan et al., 2002, Villani, 2008, Santambrogio, 2016, Chizat and Bach, 2018, Zhang et al., 2018, Martin et al., 2020].

The proximal gradient scheme that we are interested in, known as a *generalized minimizing movements* scheme [De Giorgi, 1993], has the following form,

$$\eta_{k+1}^\delta \in \arg\min_{\eta \in \mathbf{W}_2} \left[\mathscr{F}_\beta(\eta) + \frac{1}{2\delta} d^2_{\mathbf{W}_2}(\eta, \eta_k^\delta) \right] \tag{4.10}$$

where $\delta > 0$ is the time discretization length and η_k^δ is the discrete-time approximation of a return measure at time $t = k\delta$ for $k \in \mathbf{N}$. Jordan et al. [2002] presents an interpolation of this scheme that converges to the gradient flow as $\delta \to 0$. The proof relies heavily on the geometry of $\mathbf{W}_2$ – in particular, it is crucial that $\mathbf{W}_2$ is a *geodesic space* [Villani, 2008, Ambrosio et al., 2008]. This means that the 2-Wasserstein distance between any two points in $\mathbf{W}_2$ is equal to the minimum among the lengths of all curves between these points[3] measured with respect to the metric derivative. These minimizing curves are called *geodesics*, and geodesics are known to have constant speed (up to time reparameterization) [Santambrogio, 2015]. To interpolate the curve $\left(\eta_\tau^\delta\right)_{\tau \geq 0}$ as suggested by Jordan et al. [2002], we simply connect consecutive points $\eta_k^\delta, \eta_{k+1}^\delta$ by the constant speed geodesic between them, resulting in

$$\eta_{k\delta+s}^\delta = \left(\frac{\delta - s}{\delta} \mathsf{id} + \frac{s}{\delta} f_\delta^{\bar{g},\gamma} \right)_\sharp \eta_{k\delta}^\delta = \left(\mathsf{id} + s \frac{f_\delta^{\bar{g},\gamma} - \mathsf{id}}{\delta} \right)_\sharp \eta_{k\delta}^\delta \tag{4.11}$$

for $s \in (0, \delta)$. This is illustrated in Figure 4.3.

Equation (4.11) can be interpreted by imagining η as a collection of particles moving along constant speed (geodesic) paths, and the velocity of a particle z is $\delta^{-1}(f_\delta^{\bar{g},\gamma}(z) - z)$. By Theorem 4.3, the particle velocity converges so as to satisfy the distributional HJB equation (3.6) as $\delta \to 0$.

As shown in Jordan et al. [2002], the time-discretized sequence $\left\{\eta_k^\delta\right\}_{k=1}^\infty$ defined by (4.11)

[3] It is imperative that this minimum exists, which is the case in $\mathbf{W}_2$ [Villani, 2008].

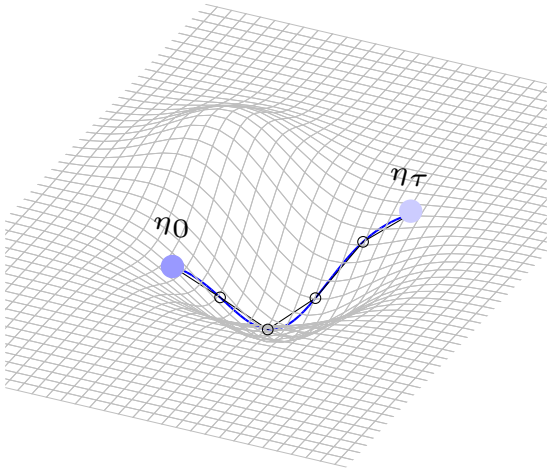


Figure 4.3: Trajectory of the return distribution in $\mathbf{W}_2$
The blue curve depicts the trajectory of the return distribution in $\mathbf{W}_2$. The piecewise linear curve (shown in black) with vertices along the trajectory illustrates how we discretize and interpolate the trajectory in time.

converges to the desired $(\eta_\tau)_{\tau \geq 0}$ as $\delta \to 0$. Thus, we can think of the operators $\mathscr{T}_\delta^\pi$ as distributional Bellman operators that can be used to approximate continuous-time policy evaluation iteratively. Note that this scheme will converge even as difference in time between successive iterates of the return distribution function tends to 0 (for instance if the parameter τ of $(\eta_\tau)_{\tau \geq 0}$ is equivalent to the flow of time in an episode rollout), so we can use this scheme to approximately compute distributional Bellman updates continuously in time.

4.3 Optimal Control

So far, we have only discussed how we can learn to *evaluate* a policy, but not how to *improve* a policy to discover a good one. Unfortunately, even in discrete time, there is no known distributional RL algorithm that converges to an optimal policy [Bellemare et al., 2017a]. On the bright side, in discrete time, there exists a policy optimization scheme for which the *mean* of the return distributions converges to the mean return.

Despite this gloomy result, existing distributional RL algorithms perform "greedy" temporal difference learning in a manner similar to Q-Learning. For instance, let $\Pi^\star$ denote the set of optimal policies, defined according to,

$$\Pi^\star = \left\{ \pi^\star \in \Pi : \forall x,\ V^{\pi^\star}(x) \geq \sup_\pi V^\pi(x) \right\}$$

where π is the set of admissible policies and $V^\pi : \mathcal{X} \to \mathcal{R}$ is the value function corresponding to the policy π. The general approach to distributional optimal control involves performing updates to minimize $d(\eta, \eta^{\pi^\star})$ for estimates of $\pi^\star \in \Pi^\star)$ in some metric[4] d. It isn't entirely surprising that only the means of the return measures converge, since the optimality condition is defined solely in terms of that statistic. However, there is no clear general alternative for comparing probability measures.

The work of Dabney et al. [2018b] proposes an alternative ordering based on *distortion risk measures*. In their IQN algorithm, return measures are represented as quantile functions, and they are compared by the mean of the return measure convolved by a function that effectively weighs its quantiles. The function can be tuned in such a way that optimal policies are more risk-averse or risk-seeking, however there is still no guarantee of the convergence to an optimal return distribution.

4.4 Summary

In this chapter, we discussed approximation schemes that can be used to perform tractable approximate policy evaluation in a convergent manner. In particular, Corollary 4.1 demonstrates a simple method of representing return measures in finite space, and equations (4.10) and (4.11) describe a discrete-time scheme that converges to a particular gradient flow. Theorem 4.3 shows that this gradient flow has a unique stationary point, which is satisfied by the ground truth return measure function. We have yet to study a concrete distributional RL algorithm, however. The following chapter presents a framework for designing distributional RL algorithms based on the tools from this section.

[4] In practice, some algorithms use functions d that are not true metrics, like f-divergences for example.

5

DEICIDE: A Framework for Distributional Reinforcement Learning of Itô Diffusions

In the previous chapter, we derived an update rule for return measures in which all return measures undergoing the update converge to a unique fixed point, which satisfies the distributional HJB equation (3.6). In this chapter, we transform this update rule to a concrete algorithm and demonstrate its effectiveness against some benchmarks.

For the remainder of the thesis, we'll consider a Feller-Dynkin process $(\mathcal{X}, \mathcal{A}, (P_t)_{t\geq 0}, r, \gamma)$ for a finite (discrete) action space $\mathcal{A}$ with respect to the probability space $(\Omega, \mathcal{F}, \Pr)$ defined previously as well as the canonical filtration $(\mathcal{F}_t)_{t\geq 0}$. Moreover, we let assumptions 3.2, 3.3, and 3.4 hold. Notably, assumption 3.4 constrains the state process $(X_t)_{t\geq 0}$ to the class of Itô Diffusions; though by the Martingale Representation Theorem this turns out to be quite robust. This chapter describes a framework for designing distributional RL algorithms in this setting, called ***D**istributional **E**valuation of **I**mplicitly-**C**ontrolled **I**tô **D**iffusion **E**volutions*, or more compactly, DEICIDE.

Moving forward, we will derive tractable, convergent approximation algorithms for continuous-

time distributional RL in §5.1. Given these algorithms, we demonstrate their results on various benchmarks in §5.2.

5.1 Algorithms

This section will give a description of DEICIDE algorithms for which we provide numerical experiments in the sequel.

We will proceed as follows:

- Based on the effects of spatial and statistical diffusivity due to imputation strategies as discussed in §4.1, we will more concretely describe a tractable method of representing return distributions in §5.1.1;
- In §5.1.2, we discuss how to compute unbiased gradient estimates in order to minimize the loss functional (4.9) governing the policy evaluation gradient flow;
- We discuss strategies for exploration and optimal control in §5.1.3;
- Finally, based on the discussions of §5.1.1, §5.1.2, and §5.1.3, in §5.1.4 we present pseudocode for two concrete continuous-time distributional RL algorithms.

5.1.1 Modeling the Return Measure Function

In order to approximate probability measures, we will employ statistical functionals and imputation strategies as discussed in §4.1. More specifically, the gradient flow optimization described by (4.10) is well-suited to "particle approximations" of probability measures – that is, measures of the form

$$\hat{\eta}(z \mid x) = \frac{1}{N}\sum_{i=1}^{N} \delta_{Z_i(x)} \qquad Z_i(x) \sim \eta(\cdot \mid x),\ i \in [N] \tag{5.1}$$

The quantities being modeled are the Dirac locations $Z_i(x)$ in (5.1), which notably can be interpreted as samples from the target measure $\eta(\cdot \mid x)$. Moreover, it is easy to verify that the quantiles of $\hat{\eta}(z \mid x)$ are precisely the quantities $Z_i(x)$, so we will simply approximate measures by their $\hat{\tau}_i$-quantiles as statistical functionals, and the imputation strategy Φ given by

$$\Phi(\{s_i\}_{i=1}^N) = \frac{1}{N}\sum_{i=1}^N \delta_{s_i} \tag{5.2}$$

While this representation is not Bellman-closed, it is approximately Bellman-closed (so approximation error in TD-like updates decays to 0 as N increases), it is simple to model, and it has demonstrated great empirical success [Dabney et al., 2018a].

Concretely, in the tabular setting, this representation requires a tensor of dimension $|\mathcal{X}||\mathcal{A}|N$. Thus, it requires N times the memory of a similar expected value RL algorithm. If we venture into the world of function approximation, we represent the statistical functionals by a set of function approximators $\{Q_i^\theta\}_{i=1}^N$, where Q_i^θ approximates $q_{\eta(\cdot|x)}(\hat{\tau}_i)$ and θ is the set of parameters. Implementing the function approximator with a neural network that shares parameters among the quantile functions until the final layer (a N-headed neural network), we again incur an N-fold memory increase.

5.1.2 Learning Return Measures

As foreshadowed, the return measures are learned by optimizing $\mathscr{F}_\beta$ as defined in (4.10). Recall that the fixed point of this optimization only satisfies (4.4) in the limit as $\beta \to 0$. We treat β as a hyperparameter.

Optimizing $\mathscr{F}_\beta$ via standard gradients is difficult, however, for a couple of reasons.

Biased stochastic Wasserstein gradients As demonstrated by Bellemare et al. [2017b], estimating gradients of the Wasserstein distances from samples are statistically biased with high probability. Although Dabney et al. [2018a] derives a method to minimize the 1-Wasserstein distance in an unbiased manner from samples, our optimization deals with the 2-Wasserstein distance.

Non-differentiability of the loss The entropy term $\mathcal{H}$ is not differentiable everywhere, so gradients cannot blindly be computed. The Wasserstein Proximal Gradient algorithm [Salim et al., 2020] accounts for this with a forward-backward Euler discretization scheme.

The irregularity of the approximated measures The measures we impute with Φ is by no means absolutely continuous with respect to the Lebesgue measure. Additionally, we know that the stationary solution of the gradient flow (4.9) is Gibbs measure with density proportional to $e^{-\phi(\cdot)}$, which has no atoms. The imputed return measure *only* have

atoms. Since (4.9) can quickly be reformulated as $D_{\mathrm{KL}}\left(\eta \parallel e^{-\phi}\right)$, the loss will always be infinite. Fortunately, this problem has already been accounted for as well. The Stein Variational Gradient Descent (SVGD) algorithm [Liu and Wang, 2019] is able to circumvent this issue by projecting the measure η to a reproducing kernel Hilbert space (RKHS) for the purpose of the update. Consequently the algorithm is biased, as we likely don't know which RKHS, if any, the return measure is part of. Alternatively, Cuturi [2013] proposes the use of the *Sinkhorn algorithm* [Sinkhorn, 1967] to directly compute $\mathscr{F}_\beta$. The Sinkhorn algorithm is itself an iterative algorithm, which suggests that it may be slow when paired with an iterative algorithm like SGD. However, Cuturi [2013] shows that this algorithm happens to be quite fast, and Martin et al. [2020] employs this algorithm successfully for the purpose of learning value distributions in discrete time.

In our experiments, we use the SVGD algorithm during the optimization procedure, in a similar manner to Zhang et al. [2018].

5.1.3 Exploration and Optimal Control

Due to the lack of theoretical guarantees surrounding return measure convergence in the optimal control setting, we follow Bellemare et al. [2017a], Hessel et al. [2018] and perform the optimization against policies with maximal means, as shown in §4.3. For the purpose of exploration, which itself is a highly complex problem, we stick to simple yet effective ϵ-greedy policies [Sutton and Barto, 2018]:

$$\pi(a \mid x) = \begin{cases} \frac{1-\epsilon}{n^\star} + \frac{\epsilon}{|\mathcal{A}|} & a \in \arg\max_{a' \in \mathcal{A}} \mathbf{E}\left[\eta(\cdot \mid x, a')\right] \\ \frac{\epsilon}{|\mathcal{A}|} & \text{otherwise} \end{cases} \tag{5.3}$$

where

$$n^\star = \left| \arg\max_{a \in \mathcal{A}} \mathbf{E}\left[\eta(\cdot \mid x, a)\right] \right|$$

and $\epsilon \in (0, 1]$.

5.1.4 Quantile DEICIDE with Function Approximation

In this section, we provide pseudocode of algorithms based on DEICIDE using quantiles as statistical functionals. The difference mainly are due to the computation of the function $\mathscr{L}^\star$ as defined in (4.8).

A model-based algorithm Algorithm 3 attempts to approximate $\mathscr{L}^\star$ directly. In order to achieve this, we must learn a stochastic model of the environment (a world model). Fortunately, we have assumed that the state process is an Itô diffusion, so we know that $\Pr(X_{t+\delta} \mid X_t)$ is Gaussian-distributed for any δ. Our world model $f_\pi^\psi : \mathcal{X} \times \mathcal{A} \to \mathcal{P}_p(\mathcal{X})$ with parameters ψ is implemented as a neural network that outputs the location and scale[1] parameters of a Gaussian distribution. Additionally, to reduce complexity, we will assume the covariance matrices of the stochastic dynamics are diagonal, so $\boldsymbol{\sigma}_\pi(x) \in \mathbf{R}^d$ will simply represent the diagonal of the covariance matrix at state x. The network is trained to minimize L^2 error between the samples from the world model and observed state differences, and updates are computed by gradient descent via the reparameterization trick [Kingma and Welling, 2013]. Then, since the quantile functions are being approximated by differentiable function approximators, the gradient and Hessian terms due to the infinitesimal generator of the Itô Diffusion governing the state process can be computed, which can be done without much trouble using an autodifferentation library such as JAX [Bradbury et al., 2018].

An issue with this implementation is that optimization tends to be unstable. Note that Algorithm 3 is not actually a temporal difference learning algorithm, since the "difference" is measured instantaneously. Consequently, there is no clear separation between the quantile function and a target function of some sort, which prohibits the use of semi-gradient updates that have are ubiquitous in the RL literature [Sutton and Barto, 2018].

A finite differences algorithm We also consider a model that is more stable under differential optimization methods. Rather than computing the actual gradient and Hessian of the quantile functions, we may employ a *finite differences* scheme to approximate these computations. Since the characterization of each quantile function of the return distribution (4.4) has precisely the same structure as the HJB equation, we apply a finite differences computation proposed by Munos [2004]. A byproduct of this scheme is that we now get to compute updates based on temporal differences, allowing us to stabilize the updates by computing semi-gradients. This is shown in Algorithm 4.

[1] Rather, we output the logarithm of the scale to ensure that the scale values are positive

Algorithm 3 Model-Based Q-DEICIDE

for each environment step k and corresponding state transition (x, a, r, x') **do**
  $\mathbf{Q}^\theta \leftarrow \mathbf{q}_{\eta(\cdot|x)} = \{q_{\eta(\cdot|x)}(\hat{\tau}_i) : i \in [N]\}$; ▷ Quantiles for return distribution at current state
  $\eta_k \leftarrow \Psi(\mathbf{Q}^\theta)$;
  $\hat{f}_\pi \leftarrow \delta^{-1}(x' - x)$; ▷ Dynamics estimate
  $(f_\pi, \log \boldsymbol{\sigma}_\pi) \leftarrow f_\pi^\psi(x, a)$;
  $\tilde{f}_\pi^\psi \leftarrow F \sim \mathcal{N}(f_\pi, \boldsymbol{\sigma}_\pi)$; ▷ Sample from world model
  $\mathcal{L}_f(\psi) \leftarrow \frac{1}{2}(\hat{f}_\pi - \tilde{f}_\pi^\psi)^2$;
  $\psi \leftarrow \psi - \alpha_\psi \nabla_\psi \mathcal{L}_f(\psi)$; ▷ Gradient computed by reparameterization trick
  for $i \in \{1, \ldots, N\}$ **do**
    $\mathscr{H}(q_i) \leftarrow r + q_i \log \gamma + \langle \nabla_x q_{\eta(\cdot|x)}(\hat{\tau}_i), f_\pi \rangle$;
    $\psi(q_i) \leftarrow \frac{1}{2}\left(\mathscr{H}(q_i) + \frac{1}{2}\boldsymbol{\sigma}_\pi^\top \mathsf{H}_x q_{\eta(\cdot|x)}(\hat{\tau}_i)\boldsymbol{\sigma}_\pi\right)^2$;
  end for
  $\widetilde{\varrho} \leftarrow e^{-\phi(\cdot)}$; ▷ Unnormalized target distribution
  $\widehat{\mathbf{Q}^\theta} \leftarrow \mathbf{Q}^\theta - \delta \mathsf{SVGD}_\theta(D_{\mathrm{KL}}(\eta_k \parallel \widetilde{\varrho}), \kappa)$; ▷ WGF KL step with kernel κ [Liu and Wang, 2019]
  $\theta \leftarrow \theta - \nabla_\theta d^2_{\mathbf{W}_2}(\Psi(\widehat{\mathbf{Q}^\theta}), \eta(\cdot \mid x))$; ▷ WGF 2-Wasserstein trust region step [Zhang et al., 2018]
end for

5.2 Experiments

5.2.1 A Stochastic Extension of Munos' Toy Problem

We begin by testing DEICIDE in a very simple environment, which is a slight modification of the toy example presented by Munos [2004]. The task consists of controlling a particle on $\mathcal{X} = [0, 1]$ with actions in $\mathcal{A} = \{-1, 1\}$. The dynamics are simply $\frac{dx}{dt} = f(x, a) = a$. The reward signal is zero in the interior of $\mathcal{X}$. When the particle reaches a boundary, the episode ends and the agent is given a stochastic reward sampled from a distribution corresponding to the endpoint it reached. Specifically,

$$r(1) \sim \mathcal{N}(2, 2)$$
$$r(0) \sim \mathcal{N}(1, 1)$$

We are interested in observing how existing distributional RL algorithms perform in this environment, and if our DEICIDE algorithms perform more favorably.

As an overview, we present a bird's eye view of the return distribution function learned by both the discrete-time and continuous-time algorithms.

Algorithm 4 Model-Based Q-DEICIDE with Finite Differences

for each environment step k and corresponding state transition (x, a, r, x') **do**
  $\mathbf{Q}^\theta \leftarrow \mathbf{q}_{\eta(\cdot|x)} = \{q_{\eta(\cdot|x)}(\hat{\tau}_i) : i \in [N]\}$; ▷ Quantiles for return distribution at current state
  $\eta_k \leftarrow \Psi(\mathbf{Q}^\theta)$;
  $\hat{f}_\pi \leftarrow \delta^{-1}(x' - x)$; ▷ Dynamics estimate
  $(f_\pi, \log \boldsymbol{\sigma}_\pi) \leftarrow f_\pi^\psi(x, a)$;
  $\tilde{f}_\pi^\psi \leftarrow F \sim \mathcal{N}(f_\pi, \boldsymbol{\sigma}_\pi)$; ▷ Sample from world model
  $\mathcal{L}_f(\psi) \leftarrow \frac{1}{2}(\hat{f}_\pi - \tilde{f}_\pi^\psi)^2$;
  $\psi \leftarrow \psi - \alpha_\psi \nabla_\psi \mathcal{L}_f(\psi)$; ▷ Gradient computed by reparameterization trick
  for $i \in \{1, \ldots, N\}$ **do**
    $h \leftarrow \frac{1}{\epsilon^2} \perp \left(\boldsymbol{\sigma}_\pi^\perp \left[q_i^\theta(x + 2\epsilon\boldsymbol{\sigma}_\pi) - 2q_i^\theta(x + \epsilon\boldsymbol{\sigma}_\pi) + q_i^\theta\right]\right)$;
    $\phi(q_i) \leftarrow \frac{1}{2}(\delta r + \gamma^\delta \perp (q_i^\theta(x')) + \frac{\delta}{2} h - q_i^\theta(x))^2$;
  end for
  $\widetilde{\varrho} \leftarrow e^{-\phi(\cdot)}$; ▷ Unnormalized target distribution
  $\widehat{\mathbf{Q}^\theta} \leftarrow \mathbf{Q}^\theta - \delta \mathsf{SVGD}_\theta(D_{\mathrm{KL}}(\eta_k \parallel \widetilde{\varrho}), \kappa)$; ▷ WGF KL step with kernel κ [Liu and Wang, 2019]
  $\theta \leftarrow \theta - \nabla_\theta d^2_{\mathbf{W}_2}(\Psi(\widehat{\mathbf{Q}^\theta}), \eta(\cdot \mid x))$; ▷ WGF 2-Wasserstein trust region step [Zhang et al., 2018]
end for

As in the expected value RL case, we see that the median of the return measure functions converges nicely to the ground truth in our continuous-time algorithm, however the discrete-time algorithm is disturbed at the point of non-differentiability. More interestingly, the return measure function learned in the discrete-time algorithm has some bizarre properties:

- The distributions are not symmetric. Since the agent only receives a single reward which is Gaussian-distributed, we should expect the return measures to be Gaussian, especially near the endpoints. This is not the case at all for the discrete-time algorithm.
- The variance of the return measure vanishes very rapidly as the state moves away from the boundaries, to the point where the return measures are effectively deterministic in most of the state space. This is not the case with our continuous-time algorithm.
- Aside from the return measures *and* their medians being off, the the return measures also do not induce an optimal policy in our experiment.

We can examine some of these oddities further. Comparing the return measures estimated near the endpoints, we observe the data shown in Figure 5.2.

We see that both algorithms tend to shed variance in the interior of the state space, however DEICIDE tends to model the full distribution substantially better, as we expect from

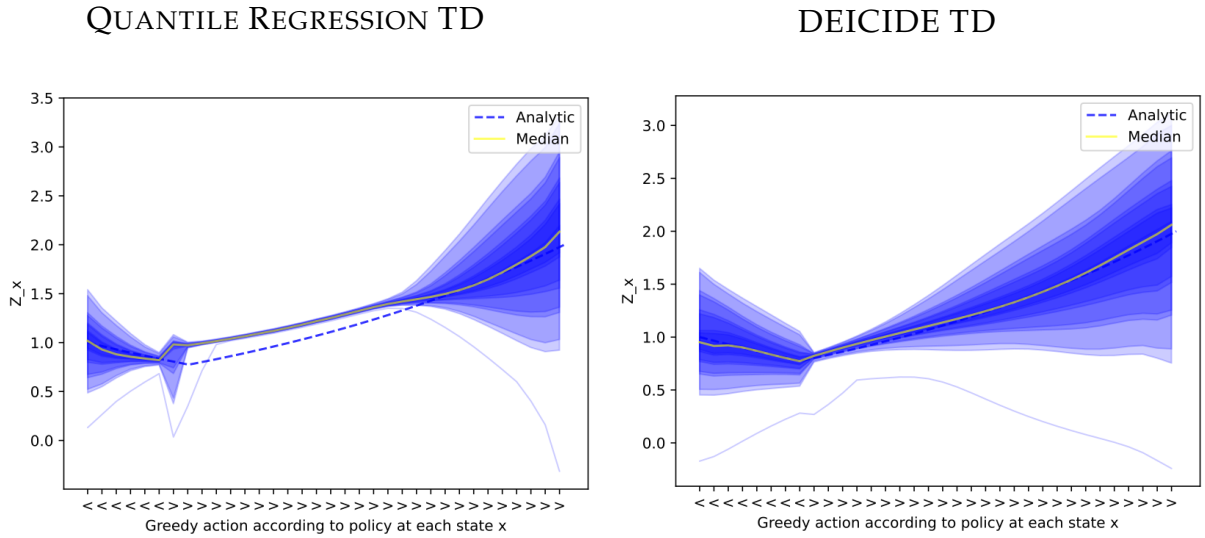


Figure 5.1: Bird's eye view of the learned return distribution functions

Figure 5.1.

5.2.2 Deterministic Environments

Recall that the analysis we presented for return distributions always assumed that the return measures are absolutely continuous with respect to the Lebesgue measure. A reasonable question to ask, then, is how DEICIDE algorithms behave in deterministic environments where the true returns have distributions $\delta_{V^\pi(\cdot)}$, since these distributions most certainly do not satisfy this assumption.

Figure 5.3 illustrates the quantiles learned by Algorithm 4 when trained on the classic CARTPOLE-V0 benchmark [Brockman et al., 2016].

We see that DEICIDE was able to accurately model the return distributions as approximate Dirac measures.

5.2.3 Deep DEICIDE

Finally, we showcase the performance of DEICIDE in a continuous-time stochastic setting. We modify the CARTPOLE-V0 benchmark [Brockman et al., 2016] to create a benchmark, which we call NOISYCONTINOUSTIMECARTPOLE-V0, as follows:

- We sample timesteps $\tau \sim \mathsf{Exponential}(100) + 10^{-3}$;
- We perturb force inputs to the cart with Gaussian noise sampled from $\mathcal{N}(0, 1)$;

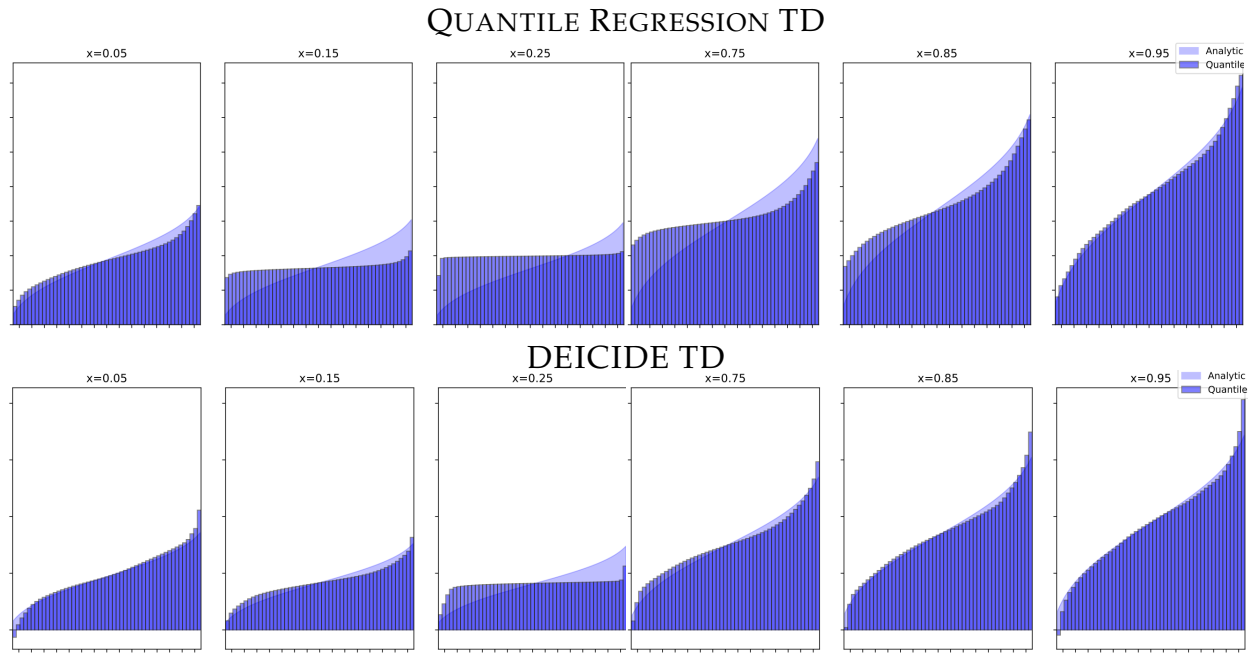


Figure 5.2: Quantile functions learned by both algorithms near the boundaries. The horizontal axis is the quantile τ and the vertical axis is the τ-quantile. The pale shaded region is the ground truth quantile function. The state input is indicated above each graph.

- We provide three more actions to deal with noise: a "NO-OP" action with applies no force, and a double force action in each direction.

We implement Algorithm 4 with a deep neural network estimating the $N = 11$ quantile functions (and another for the target quantile functions), as well as a deep neural network estimating the system dynamics and trained via the reparameterization trick.

The experiment is run over several hyperparameter configurations and random seeds, as suggested by Henderson et al. [2018]. We see that DEICIDE is fairly stable with respect to hyperparameter configurations and seeds, as illustrated in Figure 5.4.

Additionally, we see that the agents learned the optimal controller very quickly. As an example, Figure 5.5 displays the return measure function learned by the agent at a random initial state.

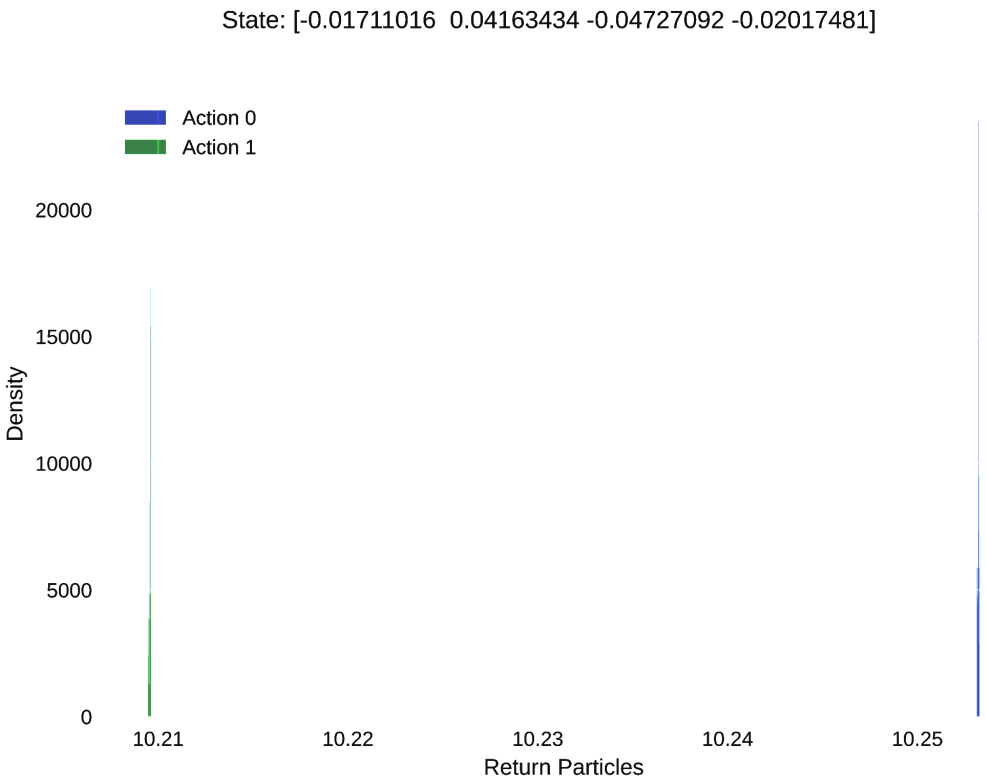


Figure 5.3: DEICIDE performance in CARTPOLE-V0

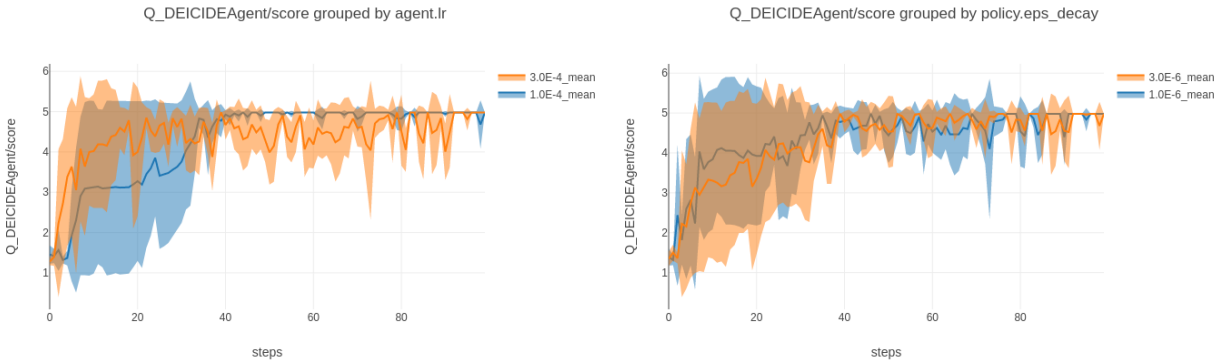


Figure 5.4: Stability of DEICIDE with respect to hyperparameter configurations and random seeds

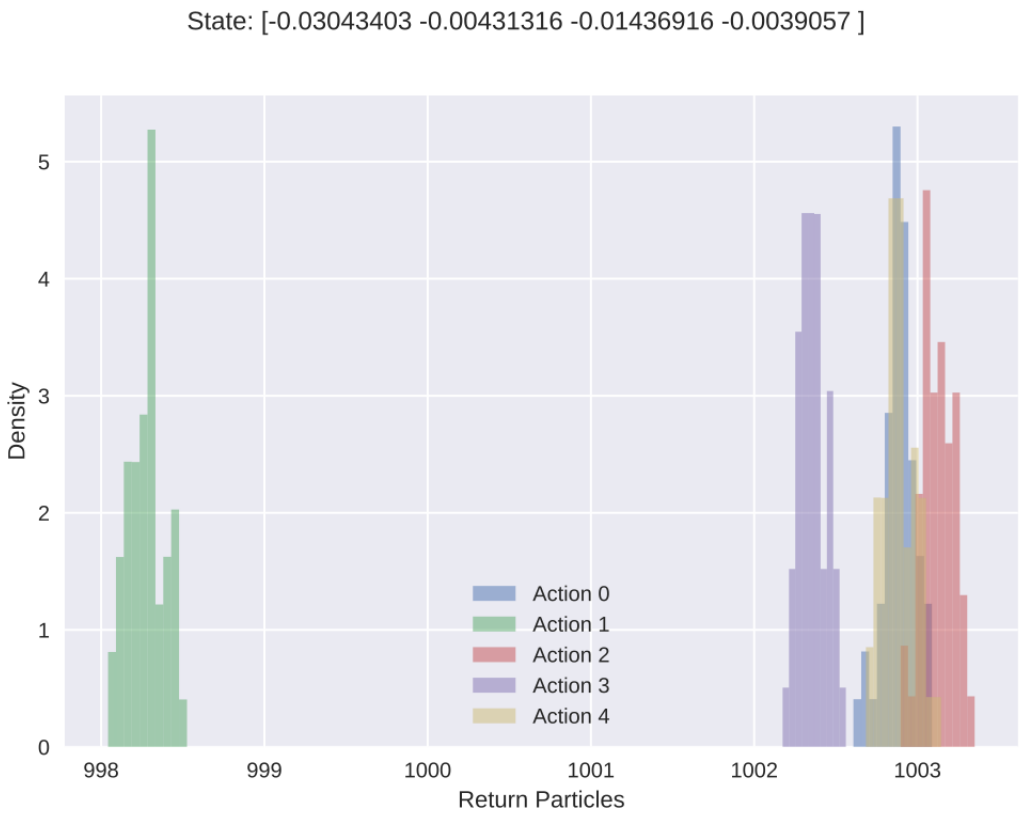


Figure 5.5: Return measure learned by a deep DEICIDE agent

6

Conclusion

In this thesis, we studied the essentially unexplored problem of learning return distributions for continuous-time Markov processes. We provide theory about the characterization of return measures in the continuous-time limit, and analyze how the tractable representation of probability measures affect this characterization.

Based on our analysis, we discuss the implementation of algorithms for continuous-time distributional reinforcement learning, and we introduce the DEICIDE framework for achieving this. Upon testing DEICIDE implementations against some simple control benchmarks, we observe that our continuous-time algorithm substantially outperforms the Quantile Regression TD learning baseline in an environment where the value function is non-differentiable, as hypothesized. In fact, the failure of discrete-time algorithms in this setting was far more pronounced in the stochastic case relative to our continuous-time implementation.

Finally, we demonstrated that DEICIDE algorithms can be endowed with highly nonlinear function approximators such as deep neural networks. We see that such implementations are able to accurately learn return distribution functions in a stochastic extension of the common CARTPOLE-V0 benchmark with randomly-sampled timesteps.

To conclude, the results presented in this thesis show promise for the prospects distributional reinforcement learning in continuous time, however lots of room is left for future work in this field. For instance, a distributional extension of Advantage Updating [Baird III, 1993] is by no means a trivial task, but may drastically improve continuous-time distributional RL algorithms. Moreover, an interesting future direction in this line of work would involve studying methods of simulating and evaluating the continuous-time performance of RL algorithms. In this thesis, we very briefly touch on this with our experiments that sample random timesteps. However, the distribution of the timestep duration was essentially arbitrary and may not be a good model for such phenomena in the real world, such as randomly-timed observations from robotic systems with several sensors. Further investigation into such models can potentially reveal further challenges in continuous-time RL that could hinder the performance of RL-trained systems in the real world.

A

A Primer on Topology

Concepts from the field of topology are mentioned often in this thesis. Some definitions and basic results are stated here.

Definition 19 (Topological Space)**.** A **topological space** is a pair $(X, \mathcal{O})$ consisting of a set X and a collection $\mathcal{O} \subset 2^X$ of subsets of X such that

1. $\emptyset \in \mathcal{O}$ and $X \in \mathcal{O}$;
2. If $(U_\alpha)_{\alpha \in I} \subset \mathcal{O}$, then $\bigcup_{\alpha \in I} U_\alpha \in \mathcal{O}$;
3. If N is a finite integer and $\{U_i\}_{i=1}^N \subset \mathcal{O}$, then $\bigcap_{i \in [N]} U_i \in \mathcal{O}$.

The set $\mathcal{O}$ is called a **topology** on X, and the elements of $\mathcal{O}$ are called **open sets**. ▽

It is only natural to ask what it means for a set to be closed.

Definition 20 (Closed Set)**.** Let $(X, \mathcal{O})$ be a topological space. A set $F \subset X$ is said to be **closed** if its complement is an open set. ▽

Remark A.1. It should be noted that openness and closedness are not mutually exclusive properties of sets – in fact, by the very definition of a topology, the "whole space" and the empty set must both be simultaneously open and closed. Such sets are called **clopen**.

The choice of the topology characterizes what it means for a function to be continuous and what it means for a sequence to converge (among other things).

Definition 21 (Continuous Function)**.** Let $(X, \mathcal{O}), (Y, \mathcal{U})$ be topological spaces. A function $f : (X, \mathcal{O}) \to (Y, \mathcal{U})$ is said to be continuous if its preimage of every open set $U \subset Y$ is an open set in X. That is,

$$U \in \mathcal{U} \implies \{x \in X : f(x) \in U\} \in \mathcal{O}$$

▽

Definition 22 (Convergence)**.** Let $(X, \mathcal{O})$ be a topological space. A sequence $\{x_i\}_{i=1}^{N} \subset X$ is said to **converge** to a point $x \in X$ if for every open set $U \ni x$ there exists a finite integer N such that $\{x_i\}_{i=N}^{\infty} \subset U$. ▽

The following proposition can be verified directly.

Proposition 4 (The Universal Topology)**.** *Let* $X = \mathbf{N}$ *and let*

$$\mathcal{O} = \{\emptyset, X, U, X \setminus U\}$$

where $U = \{4, 8, 15, 16, 23, 42\}$*. Then any sequence* $\{x_i\}_{i=1}^{\infty} \subset X$ *such that* $\{x_i\}_{i=N}^{\infty} \subset U$ *converges to* 42*, where* N *is a finite integer. For instance, the sequence* $15, 16, 15, 16, \cdots \to 42$*.*

A.1 Metric Spaces

Proposition 4 should be a little alarming. Indeed, many topological spaces are quite pathological. Usually we restrict our interests to spaces with a little more structure, such as a spaces that can be equipped with a meaningful notion of distance.

Definition 23 (Metric Space)**.** A **metric space** is a pair (X, d_X) where X is a set and $d_X : X \times X \to \mathbf{R}_+$, called a **metric** or a distance function, satisfies

1. (*Separation of points*) For any $x, y \in X$, $d_X(x, y) = 0 \iff x = y$;
2. (*Symmetry*) For any $x, y \in X$, $d_X(x, y) = d_X(y, x)$;
3. (*Triangle inequality*) For any $x, y, z \in X$, $d_X(x, z) \leq d_X(x, y) + d(y, x)$.

▽

A metric space is a special case of a topological space, where the topology is understood to be the smallest topology[1] containing all open balls $B_r(x) = \{y \in X : d_X(x, y) < r\}$. Notably, not every topological space has a metric structure. For instance, there is no function on $\mathbf{N}$ with the universal topology that satisfies the metric properties.

Remark A.2. The definitions of continuity and convergence on metric spaces coincide with those that are given in standard calculus courses.

Definition 24 (Cauchy Sequence)**.** Let (X, d) be a metric space. A sequence $\{x_i\}_{i=1}^{\infty} \subset X$ is called a **Cauchy sequence** if for every $\epsilon > 0$ there exists a finite integer N such that

$$d(x_n, x_m) \leq \epsilon \qquad \forall m, n \geq N$$

$\triangledown$

Remark A.3. Counterintuitively, Cauchy sequences may not converge. A sequence only converges (in a given topological space) if the limit lies in the space. For instance, the sequence $\{x_k\}_{k=1}^{\infty} \subset (0, 1)$ where $x_k = \frac{1}{k}$ is Cauchy, but its limiting value is 0 which is not in $(0, 1)$.

Definition 25 (Complete Space)**.** A metric space (X, d) is said to be **complete** if every Cauchy sequence in X converges in X. By the previous remark, the set $(0, 1)$ is not complete with the standard topology on the real numbers. $\triangledown$

Finally, we demonstrate a useful property of metric spaces.

Lemma A.1 (Well-behaved convergence)**.** *In any metric space (X, d), no sequence can converge to more than one point.*

Proof. Suppose $\{x_k\}_{k=1}^{\infty} \subset X$ has limits x, y. Then

$$\lim_{k \to \infty} d(x_k, x) = 0$$

[1]The *smallest* topology conforming to some constraint is the intersection of all topologies that conform to the constraint.

Moreover, by the triangle inequality, for any k we have $d(x,y) \leq d(x,x_k) + d(x_k,y)$. Therefore,

$$\begin{aligned} d(x,y) &\leq \lim_{k\to\infty} d(x,x_k) + \lim_{k\to\infty} d(x_k,y) \\ &= d(x_k,y) \end{aligned}$$

So if $x_k \to y$, then we must have $d(x,y) = 0$, and by the separation of points property this implies that $x = y$. □

B

The Basics of Measure Theory

Measure theory is a vast field of mathematical analysis that is concerned with generalizing the notion of *measure*, such as length, area, and volume, to arbitrary spaces. For the sake of building intuition, suppose we have a 3-dimensional sphere $S = \{x \in \mathbf{R}^3 : \|x\| \leq r\}$ and we are interested in measuring its volume, as well as the volume of arbitrary "pieces" of the sphere. Formally, we're looking for a function $\mathsf{Vol} : 2^S \to \mathbf{R}_+$ that maps subsets of the sphere to a non-negative real number. This function cannot just be an arbitrary function, as we expect a volume to satisfy certain properties. For instance, we need the following:

1. Emptiness has no volume: $\mathsf{Vol}(\emptyset) = 0$;
2. For disjoint subsets $A, B \subset S$, the volume of the combination of A, B should be equivalent to the sum of their original volumes: $A \cap B = \emptyset \implies \mathsf{Vol}(A + B) = \mathsf{Vol}(A) + \mathsf{Vol}(B)$;
3. For any subset $D \subset S$, no subset of D can have more volume than D: $C \subset D \implies \mathsf{Vol}(C) \leq \mathsf{Vol}(D)$.

This is not particularly interesting at first glance. However, with this definition of mea-

sure, there are some alarming consequences. In particular, the *Banach-Tarski paradox* demonstrates how one can disassemble S into a collection of pieces, and reassemble the pieces to form two identical copies of S [Banach and Tarski, 1924]. Moreover, a function that measures the length of arbitrary subsets of the real line (according to the rules above) and assigns finite length to the interval $(0, 1)$ *cannot possibly exist* [Cohn, 2013].

Interestingly, the issue lies not exactly with the rules we listed, but with the sets we wish to measure. In short, in order to construct a meaningful measure function, we must restrict this function to measure only "nice-enough" sets. In this context, the collection of "nice enough" sets is actually quite vast, and it is usually difficult to even conceive a set that is not nice enough. This will be explained in further detail in §B.1, and integration with respect to measures will be discussed in §B.2.

B.1 Measurable Spaces

The distinction of sets that can and cannot be measured is formalized by a *measurable space*. Not unlike topological spaces, a measurable space is comprised of a set with a collection of subsets, where the subsets denote the sets that one may measure. Like the topology of a set, the collection of measurable sets cannot be arbitrary. Rather, it must be a *σ-algebra*.

Definition 26 (σ-algebra)**.** Let Ω be a set. A *σ-algebra over* Ω is a collection of subsets $\Sigma \subset 2^\Omega$ such that

1. $\emptyset, \Omega \in \Sigma$;
2. $A \in \Sigma \implies A^c = \Omega \setminus A \in \Sigma$;
3. If $\{A_k\}_{k=1}^\infty$ is a countable collection of sets in Σ, then $\bigcup_k A_k \in \Sigma$.

We occasionally refer to the *smallest σ-algebra* containing a collection of subsets, or the σ-algebra *generated by* this collection of subsets. For a collection of subsets $\mathcal{U} \subset 2^\Omega$, this σ-algebra is denoted by $\sigma(\mathcal{U})$ and is defined as

$$\sigma(\mathcal{U}) = \bigcap \left\{ \Sigma \in 2^\Omega : \Sigma \text{ is a } \sigma\text{-algebra}, \quad \mathcal{U} \subset \Sigma \right\}$$

$\triangledown$

Essentially, the σ algebra describes any quantities we may want to measure. If we are

able to measure the volume of S and we are able to measure the volume of $A \subset S$, then naturally we should be able to measure the volume of $S \setminus A$. Likewise, if we can measure the volume of a countable collection of subsets of S, we should be able to measure their union. While this construction seems rather innocuous, σ-algebras can contain exceptionally "rough" sets. Below, we define a family of σ-algebras that is referred to extensively in this thesis.

Definition 27 (Borel σ-algebra)**.** Let $(X, \mathcal{O})$ be a topological space. The *Borel σ-algebra over* $(X, \mathcal{O})$ (or the Borel σ-algebra over X when the topology is implicit), denoted by $\mathscr{B}(X, \mathcal{O})$, is the smallest σ-algebra containing $\mathcal{O}$. ▽

Definition 28 (Measurable Space)**.** A *measurable space* is a pair (Ω, Σ) where Ω is a set and Σ is a σ-algebra over Ω. ▽

We are finally able to formalize the concept of a measure.

Definition 29 (Measure)**.** Let (Ω, Σ) be a measurable space. A *measure* on (Ω, Σ) is a function $\mu : \Sigma \to \mathbf{R}_+$ such that

1. $\mu(\emptyset) = 0$;
2. $A \subset B \implies \mu(A) \leq \mu(B)$;
3. If $\{A_k\}_{k=1}^{\infty}$ is a countable collection of disjoint sets in Σ (so $i \neq j \implies A_i \cap A_j = \emptyset$), then

$$\mu\left(\bigcup_{k=1}^{\infty} A_k\right) = \sum_{k=1}^{\infty} \mu(A_k)$$

A tuple (Ω, Σ, μ) is called a *measure space*. ▽

Taking a step back to the examples above, it is known that there is no measure on the measurable space $(\mathbf{R}, 2^{\mathbf{R}})$ that assigns finite measure to $(0, 1)$, and matter can be created out of thin air if we can break apart objects into any subset of space.

A very important result in measure theory is the existence of a measure space over $\mathbf{R}$ that assigns the measure $|b - a|$ to subsets of the form $(a, b), [a, b], (a, b], [a, b)$. This measure is called *the Lebesgue measure*, and it is the only measure satisfying the mentioned property. See Cohn [2013], or any textbook on measure theory, for more rigorous details.

Finally, we'll define the class of functions that preserve measure-theoretic properties.

Definition 30 (Measurability)**.** Let $(\Omega, \Sigma), (\Omega', \Sigma')$ be measurable spaces. A function f :

$(\Omega, \Sigma) \to (\Omega', \Sigma')$ is said to be *measurable* if the preimage of every Σ'-measurable set A through f is Σ-measurable. $\triangledown$

It can quickly be verified that the composition of measurable functions is itself a measurable function [Cohn, 2013]. Therefore, we can define measures through a change of variables, assuming the mapping between variables is measurable.

Definition 31 (Pushforward Measure)**.** Let (Ω, Σ, μ) be a measure space, and let (Ω', Σ') be a measurable space. For any measurable function $f : (\Omega, \Sigma) \to (\Omega', \Sigma')$, the *pushforward of f through μ*, denoted $f_\sharp \mu$, is a measure on (Ω', Σ') given by

$$f_\sharp \mu = \mu \circ f^{-1}$$

where f^{-1} is the preimage of f. $\triangledown$

B.1.1 Measure-theoretic Probability Theory

A natural application of this formalism, aside from measurements of geometric properties, is probability. In fact, we can formalize probability very easily as a measure space.

Definition 32 (Probability Space)**.** A *probability space* is a measure space (Ω, Σ, μ) where $\mu(\Omega) = 1$. $\triangledown$

Occasionally, in the context of probability, the set Ω is called the *sample space*, the σ-algebra Σ is called the *event space*, and the measure μ is called a *probability measure*.

Moreover, we can use the language of measure theory to formalize the concept of a random variables.

Definition 33 (Random Variable)**.** Let (Ω, Σ, μ) be a probability space, and let A be an arbitrary set. A *random variable* on this space is a function $Y : \Sigma \to A$. $\triangledown$

For a given measure space (Ω, Σ, μ), a property is said to hold *μ-almost everywhere* (or simply "almost everywhere" when the measure is implicit) if the property holds on all of Ω, except for possibly a set A with $\mu(A) = 0$. When μ is a probability measure, it is sometimes said that the property holds *almost surely*.

B.2 Integration

A measure can be thought of as an arbitrary method of assigning weight or density to a space. As such, the notation of integration can be formulated in terms of measures. In this section, a brief overview of this type of integration, known as Lebesgue integration, and its properties will be given.

We'll consider a measure space (Ω, Σ, μ). In order to construct an integral, we'll begin by defining the integral on a simple class of functions, aptly called the *simple functions*.

Definition 34 (Simple Function)**.** A *simple function* f is a function of the form

$$f(x) = \sum_{i=1}^{n} \alpha_i \chi_{A_i}(x)$$

where $\alpha_i \in \mathbf{R}$ and $\{A_i\}_{i=1}^n$ is a finite collection of measurable sets. $\bigtriangledown$

It is easy to verify that the sum and product of simple functions are both simple functions. The notion of integration of a simple function f with respect to a measure is fairly intuitive. We define

$$\int f(x) d\mu = \sum_{i=1}^{n} \alpha_i \mu(A_i) \qquad f = \sum_{i=1}^{n} \alpha_i \chi_{A_i}$$

By linearity, it clearly follows that the Lebesgue integral restricted to simple functions is linear. The Lebesgue integral of measurable functions f is given by

$$\int f d\mu = \sup \left\{ \int s d\mu : s \text{ is a simple function} \right\}$$

It is well known that the Lebesgue integral is linear over all measurable functions, and it is well defined. Moreover, it is known that the Lebesgue integral with respect to the Lebesgue measure agrees with the Riemann-Stieltjes integral on all integrable functions.

B.2.1 Convergence Theorems

In this section, we'll simply state some commonly known convergence properties of the Lebesgue integral. See Cohn [2013] for further details.

Theorem B.1 (The Monotone Convergence Theorem)**.** *Let (Ω, Σ, μ) be a measure space and let $\{f_i\}_{i=1}^{\infty}$ be a sequence of $[0, \infty]$-valued Σ-measurable functions. Suppose that for all $f_i \leq f_j$ for all $i \leq j$, and that $\lim_{i\to\infty} f_i(x) = f(x)$ for almost every $x \in X$. Then $\int f d\mu = \lim_{i\to\infty} \int f_i d\mu$.*

Theorem B.2 (The Dominated Convergence Theorem)**.** *Let (Ω, Σ, μ) be a measure space, g a $[0, \infty]$-valued integrable function on Ω, and $\{f_n\}_{n=1}^{\infty}$ a collection of Σ-measurable functions where $\lim_{n\to\infty} f_n(x) = f(x)$ almost everywhere. If $|f_n(x)| \leq g(x)$ almost everywhere for each n, then $\{f_n\}_{n=1}^{\infty}$ and f are integrable, and $\int f(x) = \lim_{n\to\infty} \int f_n d\mu$ almost everywhere.*

C

Tools from the Theory of Stochastic Processes

This appendix will survey some concepts from the theory of stochastic processes that are useful in the developments of this thesis. This theory tends to be quite technical, and one should be comfortable with the concepts of Appendices A and B before proceeding.

C.1 Some Special Classes of Stochastic Processes

C.1.1 Measurable, Adapted, and Progressive Processes

When dealing with stochastic processes, there are a few properties that we generally desire in order for us to be able to analyze them nicely. The most common examples will be summarized here. These definitions are due to Le Gall [2016].

For the following definitions, we will fix a probability space $(\Omega, \mathcal{F}, \Pr)$, and we will consider a stochastic process $(X_t)_{t\geq 0} \subset \mathcal{X}$, where $(\mathcal{X}, \Sigma)$ is a measurable space..

Definition 35 (Measurable Process)**.** The process $(X_t)_{t\geq 0} \subset \mathcal{X}$ is said to be *measurable* if

$(\omega, t) \mapsto X_t(\omega)$ is a measurable map on $\Omega \times \mathbf{R}_+$ with respect to the smallest σ-algebra on $\mathscr{B}(\mathbf{R}_+) \times \mathcal{F}$. ▽

For the remainder of the definitions, we will also consider a filtration (see Definition 3) $(\mathcal{F}_t)_{t\geq 0}$ making $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \Pr)$ a filtered probability space.

Definition 36 (Adapted Process)**.** The process $(X_t)_{t\geq 0} \subset \mathcal{X}$ is *adapted* if X_t is $\mathcal{F}_t$-measurable for every $t \geq 0$. ▽

Definition 37 (Progressive Process)**.** The process $(X_t)_{t\geq 0} \subset \mathcal{X}$ is *progressive* (or *progressively measurable*) if $(\omega, s) \mapsto X_t(\omega)$ is measurable on $\Omega \times [0, t]$ with respect to the smallest σ-algebra on $\mathcal{F}_t \times \mathscr{B}([0, t])$ for each $t \geq 0$. ▽

C.1.2 Martingales

Definition 38 (Martingales, Rogers and Williams [1994])**.** A **martingale** (relative to a given filtration $(\mathcal{F}_t)_{t\geq 0}$) is a stochastic process $(M_t)_{t\geq 0}$ where $M_t \in L^1$ and

$$M_s = \mathbf{E}[M_t \mid \mathcal{F}_s] \qquad 0 \leq s \leq t \tag{C.1}$$

Equation (C.1) is referred to as "the martingale property". If the equality in (C.1) is instead $\geq$ (resp. $\leq$), $(M_t)_{t\geq 0}$ is called a **supermartingale** (resp. **submartingale**). ▽

Definition 39 (Local Martingales, Le Gall [2016])**.** A **local martingale** is a stochastic process $(M_t)_{t\geq 0}$ for which there exists a sequence of nondecreasing stopping times $(T_n)_{n=1}^{\infty}$ such that $M^{T_n} = (M_{t\wedge T_n})_{t\geq 0} \in L^1$ is a martingale. ▽

Definition 40 (Semimartingales, Le Gall [2016])**.** A **semimartingale** is a random process $(X_t)_{t\geq 0}$ such that $X_t = A_t + M_t$ for each $t \geq 0$, where $(A_t)_{t\geq 0}$ is a finite variation process and $(M_t)_{t\geq 0}$ is a local martingale. ▽

C.1.3 Finite Variation Processes

Definition 41 (Finite Variation Function, Le Gall [2016])**.** Let $T \geq 0$. A continuous function $a : [0, T] \to \mathbf{R}$ with $a(0) = 0$ is said to have **finite variation** if there exists a signed measure μ on $[0, T]$ such that $a(t) = \mu([0, t])$ for any $t \in [0, T]$. ▽

A finite variation process is a process whose regularity is given by finite variation sample paths, as formalized in the next definition.

Definition 42 (Finite Variation Process, Le Gall [2016]). A process $(A_t)_{t\geq 0}$ is called a **finite variation process** if all of its sample paths are finite variation functions on $\mathbf{R}_+$. ▽

The following processes generalize the notion of covariance of random variables to stochastic processes, and appear frequently in important stochastic calculus theorems. Their definitions are given by Le Gall [2016].

Definition 43 (Quadratic Variation). Let $(M_t)_{t\geq 0}$ be a local martingale. The *quadratic variation* of $(M_t)_{t\geq 0}$, denoted $([M,M]_t)_{t\geq 0}$, is the unique increasing process such that $(M_t^2 - [M,M]_t)_{t\geq 0}$ is a local martingale. ▽

Remark C.1. The existence and uniqueness of the quadratic variation is shown by Le Gall [2016, Theorem 4.9].

Definition 44 (The Bracket of Local Martingales). Let $(M_t)_{t\geq 0}, (N_t)_{t\geq 0}$ be local martingales. The *bracket* of M, N, denoted $([M,N]_t)_{t\geq 0}$ is the finite variation process $([M,N]_t)_{t\geq 0}$ given by

$$[M,N]_t = \frac{1}{2}\Big([M+N, M+N]_t - [M,M]_t - [N,N]_t\Big)$$

▽

C.2 Itô's Lemma

Itô's Lemma is a very powerful tool in the analysis of stochastic processes. It can be thought of as a stochastic analog to Taylor's theorem.

Theorem C.1 (Itô's Lemma, Le Gall [2016]). *Let $(X^i)_{i=1}^p$ be real valued semimartingales and let $f \in C^2(\mathbf{R})$. Let $\mathbf{X}_t = (X_t^1, \ldots, X_t^p)$. Then, for every $t \geq 0$,*

$$f(\mathbf{X}_t) = f(\mathbf{X}_0) + \sum_{i=1}^{p} \int_0^t \frac{\partial f}{\partial x^i}(\mathbf{X}_s) dX_s^i + \frac{1}{2}\sum_{i=1}^{p}\sum_{j=1}^{p} \int_0^t \frac{\partial^2 f}{\partial x_i \partial x_j}(\mathbf{X}_s) d[X^i, X^j]_s \tag{C.2}$$

C.3 The Feynman-Kac Formula

We make use of the following formulation of the *Feynman-Kac formula*, as illustrated in Le Gall [2016, Exercise 6.26].

Theorem C.2. *Let $(X_t)_{t\geq 0}$ be a Feller-Dynkin process in a space $\mathcal{X}$ and let $v \in C_0(\mathcal{X})$. Define*

for any $x \in \mathcal{X}$ and ϕ a bounded and measurable function over $\mathcal{X}$ the transition semigroup $(Q_t^\star)_{t\geq 0}$ where

$$Q_t^\star \phi(x) = \mathbf{E}\left[\phi(X_t)\exp\left(-\int_0^t v(X_s)ds\right) \;\middle|\; X_0 = x\right]$$

If $(X_t)_{t\geq 0}$ admits an infinitesimal generator $\mathscr{L}$ and $\phi \in \mathcal{D}(\mathscr{L})$, then

$$\frac{d}{dt}Q_t^\star \phi|_{t=0} = \mathscr{L}\phi - v\otimes\phi \tag{C.3}$$

Remark C.2. The Feynman-Kac formula can be seen as the Kolmogorov Backward Equation with an "integrating factor". Effectively, the Feynman-Kac formula allows us to identify solutions of PDEs of the form

$$\frac{\partial u}{\partial t} = -\mathscr{L}u + v\otimes\phi$$

with conditional expectations of diffusion processes.

D

Tempered Distributions

A recurring concept in many areas of mathematics, physics, and engineering is that of *generalized functions*, known as *distributions*[1]. One such example is the Dirac delta. Distributions are particularly helpful at formally describing weakened solutions to PDEs by objects that may not be functions.

In this thesis, we will make use of the class of *tempered* distributions, whose definition will be given in this appendix. For more details, refer to Lax and Sons [2002].

Definition 45 (Schwartz Class)**.** Let X be a normed space. A *Schwartz class* is a class $\mathcal{S}$ of rapidly decaying-smooth functions,

$$\mathcal{S} = \left\{ f \in C^\infty(X; \mathbf{R}) : \sup_{x \in X} (1 + \|x\|^k) |f^{(m)}(x)| < \infty \quad \forall k, m \in \mathbf{N} \right\}$$

$\triangledown$

Definition 46 (Tempered Distribution)**.** A tempered distribution is an element of the topo-

[1] Not to be confused with probability distributions.

logical dual[2] $\mathcal{S}'$ of the Schwartz class $\mathcal{S}$. $\triangledown$

Remark D.1. The Dirac delta is the operator δ such that $\langle \delta, \phi \rangle = \phi(0)$. Clearly δ is linear, and since it is bounded, it is continuous. Therefore δ is indeed a tempered distribution.

Tempered distributions admit a notion of differentiability, which can be used to define "distributional" solutions to PDEs.

Definition 47 (Distributional Derivative)**.** Let $\mathcal{S}$ be a Schwartz class and $\psi \in \mathcal{S}'$ a tempered distribution. Then ψ has a distributional derivative if there exists a tempered distribution ψ' for which

$$\langle \psi', \phi \rangle = -\langle \psi, \phi' \rangle \qquad \forall \phi \in \mathcal{S},$$

and ψ' is called the distributional derivative of ψ. $\triangledown$

Definition 48 (Distributional Solutions of Hamilton-Jacobi PDEs)**.** Consider the following PDE,

$$\frac{\partial u}{\partial t} = f \circ u + \langle \nabla u, g \rangle + h^\top \mathsf{H}_y u h \tag{D.1}$$

where $u \in C^2(\mathbf{R}_+ \times \mathcal{Y}; \mathbf{R})$ for a normed space $\mathcal{Y}$.

Then $\psi \in \mathcal{S}'$ is said to be a *distributional solution* to (D.1) if

$$\begin{aligned} &\int_0^\infty \int_{\mathcal{Y}} \phi(t,y) \left(f(\psi(y)) - \frac{\partial}{\partial t} \psi(y) \right) dy dt \\ &\qquad = \int_0^\infty \int_{\mathcal{Y}} \Big[\langle \psi(y) g(y), \nabla_y \phi(t,y) \rangle - h(y)^\top \psi(y) \mathsf{H}_y \phi(t,y) h(y) \Big] dy dt \end{aligned}$$

for every test function $\phi \in \mathcal{S}$. This is justified by simply multiplying both sides of (D.1) by the test function, integrating over $\mathbf{R}_+ \times \mathcal{Y}$, and substituting gradient terms of ψ with respect to its distributional derivative. $\triangledown$

[2] The dual of a normed space is the set of all continuous, linear functionals on that space.

Bibliography

Alekh Agarwal, Nan Jiang, Sham M Kakade, and Wen Sun. Reinforcement learning: Theory and algorithms. *CS Dept., UW Seattle, Seattle, WA, USA, Tech. Rep*, 2019.

Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient flows: in metric spaces and in the space of probability measures*. Springer Science & Business Media, 2008.

Dilip Arumugam, Peter Henderson, and Pierre-Luc Bacon. An information-theoretic perspective on credit assignment in reinforcement learning. *arXiv preprint arXiv:2103.06224*, 2021.

Adrià Puigdomènech Badia, Bilal Piot, Steven Kapturowski, Pablo Sprechmann, Alex Vitvitskyi, Zhaohan Daniel Guo, and Charles Blundell. Agent57: Outperforming the atari human benchmark. In *International Conference on Machine Learning*, pages 507–517. PMLR, 2020.

Leemon C Baird. Reinforcement learning in continuous time: Advantage updating. In *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, volume 4, pages 2448–2453. IEEE, 1994.

Leemon C Baird III. Advantage updating. Technical report, WRIGHT LAB WRIGHT-PATTERSON AFB OH, 1993.

Stefan Banach and Alfred Tarski. Sur la décomposition des ensembles de points en parties respectivement congruentes. *Fund. math*, 6(1):244–277, 1924.

Andrew G Barto, Richard S Sutton, and Peter S Brouwer. Associative search network: A reinforcement learning associative memory. *Biological cybernetics*, 40(3):201–211, 1981.

Marc G Bellemare, Georg Ostrovski, Arthur Guez, Philip Thomas, and Rémi Munos. Increasing the action gap: New operators for reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

Marc G. Bellemare, W. Dabney, and R. Munos. A distributional perspective on reinforcement learning. In *ICML*, 2017a.

Marc G Bellemare, Ivo Danihelka, Will Dabney, Shakir Mohamed, Balaji Lakshminarayanan, Stephan Hoyer, and Rémi Munos. The cramer distance as a solution to biased wasserstein gradients. *arXiv preprint arXiv:1705.10743*, 2017b.

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.

Richard Bellman. The theory of dynamic programming. Technical report, Rand corp santa monica ca, 1954.

Richard Bellman. A markovian decision process. *Journal of mathematics and mechanics*, 6 (5):679–684, 1957.

Jean-David Benamou and Yann Brenier. A computational fluid mechanics solution to the monge-kantorovich mass transfer problem. *Numerische Mathematik*, 84(3):375–393, 2000.

Dimitri P Bertsekas and John N Tsitsiklis. *Neuro-dynamic programming*. Athena Scientific, 1996.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. Jax: composable transformations of python+numpy programs. 2018. URL http://github.com/google/jax.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

Lenaic Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. *arXiv preprint arXiv:1805.09545*, 2018.

Donald L Cohn. *Measure theory*. Springer, 2013.

Michael G Crandall and Pierre-Louis Lions. Viscosity solutions of hamilton-jacobi equations. *Transactions of the American mathematical society*, 277(1):1–42, 1983.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transportation distances, 2013.

W. Dabney, M. Rowland, Marc G. Bellemare, and R. Munos. Distributional reinforcement learning with quantile regression. In *AAAI*, 2018a.

Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. In *International conference on machine learning*, pages 1096–1105. PMLR, 2018b.

Ennio De Giorgi. New problems on minimizing movements. *Ennio de Giorgi: Selected Papers*, pages 699–713, 1993.

Ennio De Giorgi, Antonio Marino, and Mario Tosques. Problems of evolution in metric spaces and maximal decreasing curve. *Atti Accad. Naz. Lincei Rend. Cl. Sci. Fis. Mat. Natur.(8)*, 68(3):180–187, 1980.

Eric V Denardo. Contraction mappings in the theory underlying dynamic programming. *Siam Review*, 9(2):165–177, 1967.

Kenji Doya. Reinforcement learning in continuous time and space. *Neural computation*, 12 (1):219–245, 2000.

Ioannis Exarchos and Evangelos A Theodorou. Stochastic optimal control via forward and backward stochastic differential equations and importance sampling. *Automatica*, 87:159–165, 2018.

Wendell H Fleming and Halil Mete Soner. *Controlled Markov processes and viscosity solutions*, volume 25. Springer Science & Business Media, 2006.

Jerome H Friedman. *The elements of statistical learning: Data mining, inference, and prediction*. springer open, 2017.

Igor Vladimirovich Girsanov. On transforming a certain class of stochastic processes by absolutely continuous substitution of measures. *Theory of Probability & Its Applications*, 5(3):285–301, 1960.

Thomas Hakon Gronwall. Note on the derivatives with respect to a parameter of the solutions of a system of differential equations. *Annals of Mathematics*, pages 292–296, 1919.

Igor Halperin. Distributional offline continuous-time reinforcement learning with neural physics-informed pdes (sciphy rl for doctr-l). *arXiv preprint arXiv:2104.01040*, 2021.

J Michael Harrison. *Brownian models of performance and control*. Cambridge University Press, 2013.

W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. 1970.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

Juan Camilo Gamboa Higuera, David Meger, and Gregory Dudek. Synthesizing neural network controllers with probabilistic model-based reinforcement learning. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2538–2544. IEEE, 2018.

Ronald A Howard. Dynamic programming and markov processes. 1960.

Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the fokker-planck equation. *SIAM J. Math. Anal*, 29:1–17, 2002.

Mark Kac. On distributions of certain wiener functionals. *Transactions of the American Mathematical Society*, 65(1):1–13, 1949.

Jeongho Kim, Jaeuk Shin, and Insoon Yang. Hamilton-jacobi deep q-learning for deterministic continuous-time systems with lipschitz continuous controls. *Journal of Machine Learning Research*, 22(206):1–34, 2021.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Roger Koenker and Gilbert Bassett Jr. Regression quantiles. *Econometrica: journal of the Econometric Society*, pages 33–50, 1978.

Andrei Kolmogoroff. Über die analytischen methoden in der wahrscheinlichkeitsrechnung. *Mathematische Annalen*, 104(1):415–458, 1931.

Erwin Kreyszig. *Introductory functional analysis with applications*, volume 1. wiley New York, 1978.

P.D. Lax and John Wiley & Sons. *Functional Analysis*. Pure and Applied Mathematics: A Wiley Series of Texts, Monographs and Tracts. Wiley, 2002. ISBN 9780471556046. URL https://books.google.ca/books?id=-jbvAAAAMAAJ.

Jean-François Le Gall. *Brownian motion, martingales, and stochastic calculus*. Springer, 2016.

Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Long-Ji Lin. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine learning*, 8(3-4):293–321, 1992.

Long Ji Lin. Scaling up reinforcement learning for robot control. In *ICML*, 1993.

Qiang Liu and Dilin Wang. Stein variational gradient descent: A general purpose bayesian inference algorithm, 2019.

Michael Lutter, Boris Belousov, Shie Mannor, Dieter Fox, Animesh Garg, and Jan Peters. Continuous-time fitted value iteration for robust policies. *arXiv preprint arXiv:2110.01954*, 2021a.

Michael Lutter, Shie Mannor, Jan Peters, Dieter Fox, and Animesh Garg. Value iteration in continuous actions, states and time. *arXiv preprint arXiv:2105.04682*, 2021b.

John D. Martin, Michal Lyskawinski, Xiaohu Li, and Brendan Englot. Stochastically Dominant Distributional Reinforcement Learning, 2019.

John D. Martin, Michal Lyskawinski, Xiaohu Li, and Brendan Englot. Stochastically dominant distributional reinforcement learning, 2020.

Borislav Mavrin, Hengshuai Yao, Linglong Kong, Kaiwen Wu, and Yaoliang Yu. Distributional reinforcement learning for efficient exploration. In *International conference on machine learning*, pages 4424–4434. PMLR, 2019.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

R. Munos. A study of reinforcement learning in the continuous case by the means of viscosity solutions. *Machine Learning*, 40:265–299, 2004.

Rémi Munos. A convergent reinforcement learning algorithm in the continuous case based on a finite difference method. In *IJCAI (2)*, pages 826–831, 1997.

Rémi Munos and Paul Bourgine. Reinforcement learning for continuous stochastic control problems. In *NIPS*, pages 1029–1035, 1997.

Matteo Muratori and Giuseppe Savaré. Gradient flows and evolution variational inequalities in metric spaces. i: structural properties, 2018.

Marcus Pereira, Ziyi Wang, Ioannis Exarchos, and Evangelos A Theodorou. Learning deep stochastic optimal control policies using forward-backward sdes. *arXiv preprint arXiv:1902.03986*, 2019.

Jan Peters, Sethu Vijayakumar, and Stefan Schaal. Reinforcement learning for humanoid robotics. In *Proceedings of the third IEEE-RAS international conference on humanoid robots*, pages 1–20, 2003.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Martin L Puterman and Shelby L Brumelle. On the convergence of policy iteration in stationary dynamic programming. *Mathematics of Operations Research*, 4(1):60–69, 1979.

Maxim Raginsky, Alexander Rakhlin, and Matus Telgarsky. Non-convex learning via stochastic gradient langevin dynamics: a nonasymptotic analysis. In *Conference on Learning Theory*, pages 1674–1703. PMLR, 2017.

Gareth O Roberts and Osnat Stramer. Langevin diffusions and metropolis-hastings algorithms. *Methodology and computing in applied probability*, 4(4):337–357, 2002.

Ralph Tyrell Rockafellar. *Convex analysis*. Princeton university press, 2015.

L Chris G Rogers and David Williams. Diffusions, markov processes and martingales, volume 1: Foundations. *John Wiley & Sons, Ltd., Chichester*, 7, 1994.

Uwe Rösler. A fixed point theorem for distributions. *Stochastic Processes and their Applications*, 42(2):195–214, 1992.

Mark Rowland, Robert Dadashi, Saurabh Kumar, Remi Munos, Marc Bellemare, and Will Dabney. Statistics and samples in distributional reinforcement learning. In *Proceedings of the 36th International Conference on Machine Learning*, pages 5528–5536, 2019. URL http://proceedings.mlr.press/v97/rowland19a/rowland19a.pdf.

Adil Salim, Anna Korba, and Giulia Luise. The wasserstein proximal gradient algorithm. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12356–12366. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/91cff01af640a24e7f9f7a5ab407889f-Paper.pdf.

F. Santambrogio. {Euclidean, metric, and Wasserstein} gradient flows: an overview. *Bulletin of Mathematical Sciences*, 7:87–154, 2016.

Filippo Santambrogio. Optimal transport for applied mathematicians. *Birkäuser, NY*, 55 (58-63):94, 2015.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

David Silver, Satinder Singh, Doina Precup, and Richard S. Sutton. Reward is enough. *Artificial Intelligence*, page 103535, 2021. ISSN 0004-3702. doi: https://doi.org/10.1016/j.artint.2021.103535. URL https://www.sciencedirect.com/science/article/pii/S0004370221000862.

Richard Sinkhorn. Diagonal equivalence to matrices with prescribed row and column sums. *The American Mathematical Monthly*, 74(4):402–405, 1967.

William D Smart and L Pack Kaelbling. Effective reinforcement learning for mobile robots. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, volume 4, pages 3404–3410. IEEE, 2002.

Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3(1):9–44, 1988.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Richard Stuart Sutton. *Temporal credit assignment in reinforcement learning*. PhD thesis, University of Massachusetts Amherst, 1984.

Corentin Tallec, Léonard Blier, and Yann Ollivier. Making deep q-learning methods robust to time discretization. In *International Conference on Machine Learning*, pages 6096–6104. PMLR, 2019.

Yuval Tassa and Tom Erez. Least squares solutions of the hjb equation with neural network value-function approximators. *IEEE transactions on neural networks*, 18(4):1031–1041, 2007.

Gerald Tesauro. Td-gammon, a self-teaching backgammon program, achieves master-level play. *Neural computation*, 6(2):215–219, 1994.

Matthew Thorpe. Introduction to optimal transport, 2018. URL https://www.math.cmu.edu/~mthorpe/OTNotes.pdf.

Carsten A Ullrich. *Time-dependent density-functional theory: concepts and applications*. OUP Oxford, 2011.

Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

Christopher John Cornish Hellaby Watkins. Learning from delayed rewards. 1989.

Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 681–688. Citeseer, 2011.

Andre Wibisono. Sampling as optimization in the space of measures: The langevin dynamics as a composite optimization problem. In *Conference on Learning Theory*, pages 2093–3027. PMLR, 2018.

Christopher K Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

Jianyi Zhang and Paul Weng. Safe distributional reinforcement learning. *arXiv preprint arXiv:2102.13446*, 2021.

Ruiyi Zhang, C. Chen, C. Li, and L. Carin. Policy optimization as wasserstein gradient flows. In *ICML*, 2018.