# Couplings in Reinforcement Learning:
## Applications to State Abstraction and Algorithm Analysis

Philip Amortila

School of Computer Science

McGill University, Montreal

August 2019

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

## Abstract

This thesis develops aspects of the theory of reinforcement learning using the notion of probabilistic couplings. We provide a unifying framework which is suitable for demonstrating: (1) behavioural equivalence of states in Markov decision processes (MDPs), and (2) convergence and equivalence of stochastic approximation algorithms. A unifying theme is that the presence (or lack thereof) of specific couplings between state transition distributions incurs knowledge about their long-term behaviours.

The first application of our methods is the construction of *temporally extended metrics* for measuring behavioural similarity of states in MDPs. The temporally extended metrics extend a base metric between states (e.g. reward difference or value difference) so as to reflect not just the current difference but the extent to which the difference is preserved throughout the course of transitions. The construction is based on a generalized notion of bisimulation given in terms of probabilistic couplings. We provide safety bounds on the approximation error incurred when using these metrics for state abstraction.

In the second application, we propose a *distributional* perspective on stochastic approximation theory for RL by working at the level of distributions of possible function estimates. In this framework, simple couplings can be exhibited to show that many commonly-used value-based RL algorithms are *contractions* on the space of distributions, and thus that they converge to stationary distributions. We provide general criteria for convergence, and characterize the attained limit distributions. The proof methods generalize and simplify existing arguments in the literature.

## Résumé

Cette thèse développe des aspects de la théorie de l'apprentissage par renforcement (en anglais *reinforcement learning*, RL) en utilisant la notion de couplage probabiliste. Nous fournissons un cadre unificateur qui convient pour démontrer: (1) l'équivalence comportementale d'états dans les processus de décision Markovien (en anglais *Markov decision process*, MDP), et (2) la convergence et l'équivalence des algorithmes d'approximation stochastique. Un thème unificateur est que la présence (ou l'absence) de couplages spécifiques entre les distributions de transition d'état implique la connaissance de leurs comportements à long terme. La première application de nos méthodes est la construction de métriques étendues dans le temps (en anglais *temporally extended metrics*) pour mesurer la similarité comportementale d'états dans les MDP. Les métriques étendues dans le temps étendent une métrique de base entre états (par exemple, une différence de récompense ou une différence de valeur) de manière à refléter non seulement la différence actuelle, mais également la mesure dans laquelle la différence est préservée au cours des transitions. La construction est basée sur une notion généralisée de bisimulation donnée en termes de couplages probabilistes. Nous fournissons des bornes de sécurité sur l'erreur d'approximation lors de l'utilisation de ces métriques pour l'abstraction d'état. Dans la seconde application, nous proposons une perspective distributionnelle de la théorie de l'approximation stochastique en travaillant au niveau des distributions d'estimations de fonction possibles. Dans ce cadre, des couplages simples peuvent être présentés pour montrer que de nombreux algorithmes RL basés sur l'estimation des valeurs sont des contractions sur l'espace des distributions et donc convergent vers des distributions stationnaires. Nous fournissons des critères généraux de convergence et caractérisons les distributions limites atteintes. Les méthodes de preuve généralisent et simplifient les arguments existants dans la littérature.

## Acknowledgements

It goes without saying that this thesis would not have been possible without the help of my supervisors Prakash Panangaden, Marc G. Bellemare, and Doina Precup. I was very fortunate to have received the attention of three great advisers and researchers. It was inspiring and educational to have seen them in action, and as a result I have grown tremendously as a researcher. Thanks to Prakash, Marc, and Doina for providing me with interesting and fun research topics and for following me through the many tangents I took during the course of exploration.

I am also grateful to Guillaume Rabusseau, a great mentor, colleague, and friend. Guillaume generously provided me with guidance, showed me the ropes of graduate school, taught me how to take graduate school seriously, and, occasionally, taught me how to not take it too seriously. Thanks also to Audrey Durand for always being available for existential discussions.

I would also like to thank the many great friends in and outside the lab which I have been lucky to know and have made the past two years so special.

Thanks most of all to my parents, who have provided me with countless opportunities and endless support over the past two years and also over the past twenty-five.

## Contribution of authors

Most of Chapter 3 also appears in the workshop paper:

- P. Amortila, M. G. Bellemare, P. Panangaden, and D. Precup (2018) ”Temporally Extended Metrics for Markov Decision Processes”. SafeAI Workshop at AAAI Conference on Artificial Intelligence 2019.

Chapters 4 and 5 contain new material thus far not published. All chapters are joint work between the author and his supervisors Prakash Panangaden, Marc G. Bellemare, and Doina Precup.

*To 3705, 4442, 3621, 3494, and the people therein*

# Contents

**1 Introduction** **4**

1.1 Contributions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

**2 Background** **9**

2.1 Markov Decision Processes . . . . . . . . . . . . . . . . . . . . . . . 9

2.2 Bellman Equations and Dynamic Programming . . . . . . . . . . . . . . . . 13

2.3 Value-based Reinforcement Learning . . . . . . . . . . . . . . . . . . . 14

2.4 Optimal Transport and the Coupling Method . . . . . . . . . . . . . . . . . 16

2.5 State Abstractions and Bisimulation Relations . . . . . . . . . . . . . . . . 19

2.6 Bisimulation metrics . . . . . . . . . . . . . . . . . . . . . . . . . 21

**3 Temporally Extended Metrics: State Abstraction via Couplings** **23**

3.1 Bisimulation via Liftings . . . . . . . . . . . . . . . . . . . . . . . . 24

3.2 Φ-bisimulation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 28

3.3 Temporally Extended Metrics . . . . . . . . . . . . . . . . . . . . . . 30

3.4 Comparing Bisimulation Metrics and Temporally Extended Metrics . . . . 33

3.4.1 Bounds . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 33

3.4.2 Optimal Couplings . . . . . . . . . . . . . . . . . . . . . . . . 35

3.5 Related Work . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 38

3.6 Discussion & Future Work . . . . . . . . . . . . . . . . . . . . . . . . . . 39

**4 Bisimulation of algorithms: Distributional RL vs. Expected RL 41**

4.1 Distributional RL . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 42

4.2 Value-based RL algorithms define Markov Processes on the Space of Value Functions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 43

4.3 Bisimilarity of Distributional RL and Expected RL . . . . . . . . . . . . . . 44

4.4 Discussion & Future work . . . . . . . . . . . . . . . . . . . . . . . . . . . 45

**5 Distributional Stochastic Approximation: Convergence via Couplings 47**

5.1 A general convergence criterion . . . . . . . . . . . . . . . . . . . . . . . 49

5.1.1 Proof strategy . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 52

5.2 Examples: Monte Carlo, SARSA, Q-Learning, TD($\lambda$) . . . . . . . . . . . . . 52

5.2.1 Monte Carlo Evaluation . . . . . . . . . . . . . . . . . . . . . . . 53

5.2.2 TD(0) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 53

5.2.3 SARSA . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 54

5.2.4 Q-Learning . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 54

5.2.5 TD($\lambda$) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 55

5.3 Characterizing the stationary distributions . . . . . . . . . . . . . . . . . . 56

5.3.1 Mean is the true value function . . . . . . . . . . . . . . . . . . . 57

5.3.2 Covariance . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 57

5.4 Related Work . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 60

5.5 Discussion & Future Work . . . . . . . . . . . . . . . . . . . . . . . . . . 61

**6 Conclusion 64**

# Chapter 1

# Introduction

Modern machine learning inevitably deals with stochasticity: data is generated by sampling from an unknown distribution and the analysis of said data is often done by stochastic algorithms. In reinforcement learning (RL), we further deal with the problem of sequential decision-making in a changing and stochastic environment. Thus, reasoning about uncertainty and the behaviour of stochastic transition systems lies at the heart of RL.

In this thesis we develop some aspects of the theory of RL using the notion of *probabilistic coupling*. The coupling method is a long-studied and invaluable tool in probability theory, where it exhibits a wealth of applications and a rich literature (Lindvall 2002; Thorisson 2000). Couplings have recently received some attention in the broader machine learning literature via applications of optimal transport and the Wasserstein metric (Arjovsky, Chintala, and Bottou 2017; Bellemare, Dabney, and Munos 2017). However, these are mainly introduced as metrics to be optimized, and the deeper properties of couplings (which makes them useful as a proof strategy) are rarely exploited. Loosely, a proof by coupling consists of comparing two stochastic processes by correlating their sources of randomness and thus simplifying deductions about the original processes. Paraphrasing from Hsu: proofs by coupling simplify probabilistic reasoning by reducing to one source of randomness, abstracting away probabilities, and enabling compositional, structured reasoning (Hsu 2017). Indeed, proofs by coupling have recently emerged in the differential privacy and formal verification communities (Barthe et al. 2016), where they have simplified existing arguments that were often subtly incorrect (Ding et al. 2018). This thesis shows that similar room for improvement exists in the RL literature: proofs of convergence (primarily based on stochastic approximation theory) are often elaborate and intricate, and furthermore have to be tailored to individual algorithms. We suggest that coupling methods can provide a unifying framework and simplify existing results. Namely, we show that the existence of a particular coupling between two arbitrary initializations of the same algorithm is enough to enable proofs of convergence for many commonly-used RL algorithms (cf. Section 5).

We demonstrate the effectiveness of couplings in RL by showing that the existence of specific couplings can directly imply relational properties about states in Markov Decision Processes (to be used for state abstraction) or even about different RL algorithms (to be used for convergence analyses). More concretely, we provide a single theoretical framework which can be used to show: (1) *behavioural equivalence* of states in MDPs (via bisimulation-like properties), (2) *behavioural equivalence* of RL algorithms, and (3) convergence guarantees of RL algorithms (via contraction arguments on the distribution of possible iterates). We now provide, in more detail, an overview of these 3 contributions.

## 1.1 Contributions

**Temporally Extended Metrics: State Abstraction via Couplings (Ch. 3)**

If two states of an MDP are related in some way, how can we verify that, after performing any number of transitions, the new states will continue to be related? In this chapter we formalize such a notion of state equivalence based on couplings, which can be defined for any given relation of interest. We discuss its applications to *state abstraction* methods, by considering relations based on (approximate) reward equality or (approximate) value equality of states. A long-studied notion for state abstractions in a Markov Decision Process (MDP) is called bisimulation. There also exists quantitative analogues called bisimulation metrics. However, these metrics are prohibitively expensive to compute which has limited their applicability. We instead propose alternative metrics for behavioural equivalence by developing a notion of *temporally extended metrics*, which extend a base metric between states of an environment so as to reflect not just the current difference but the extent to which the distance is preserved through the course of transitions. We further show that this property is not satisfied by the bisimulation metrics. The construction relies on a generalized notion of bisimulation relations, which is based on couplings and considers arbitrary comparisons between states instead of strict reward matching. A temporal extension can be defined for any base metric of interest which makes the construction very flexible. The kernel of the temporally extended metrics corresponds precisely to exact bisimulation, thus assigning distance $0$ only to states which are indistinguishable. We provide bounds relating bisimulation and temporally extended metrics and also examine the couplings of state distributions which are induced.

### Bisimulation of algorithms: Distributional RL vs. Expected RL (Ch. 4)

The distributional family of reinforcement learning algorithms have produced state-of-the-art empirical results, however the theoretical properties responsible for the improvements over their expected-value counterparts are not clearly understood. In this short section, we use the tools developed in Chapter 3 to establish the equivalence (in expectation) of the distributional family of reinforcement learning algorithms with their expected-value counterparts. The bridge from the previous work comes from casting these algorithms as Markov processes in their own right by considering the time-evolution of the *distribution of possible value function estimates* induced by the possible samples of the algorithms. In this framework one can then use the same coupling-based techniques. In particular we formalize a notion of *bisimulation for algorithms*, which can be used to determine if two algorithms are behaviourally identical. We verify that, in the tabular setting, the distributional and expected versions of the SARSA algorithm are *bisimilar*. These results rephrase a recent paper by (Lyle, Castro, and Bellemare 2019). However, we wish to emphasize the novel use of couplings and bisimulation techniques for the analysis of these algorithms. Bisimulation, in particular, has previously only been viewed as a relation for state abstraction of states in MDPs.

### Distributional Stochastic Approximation: Convergence via Couplings (Ch. 5)

Our last contribution establishes convergence in distribution of a wide class of commonly used value-based RL algorithms when using synchronous updates and constant step-sizes. We exploit the Markov process view described in Chapter 4, and work at the level of *distribution of possible value function estimates*. We show that many common RL algorithms induce Markov kernels which are *contractions* on the space of distributions. This property holds true whenever a certain coupling can be constructed. Our proofs by coupling are simpler than existing stochastic approximation-based methods, showing promise that the methods can be extended to further analyses. We provide general criteria guaranteeing contraction (and thus convergence to a stationary distribution), and further provide a characterization of the attained stationary distribution.

# Chapter 2

# Background

In reinforcement learning (RL), we consider an *agent* interacting with an *environment* to accomplish some high-level task, and wish for the agent to arrive at some understanding of which behaviour to adopt in order to optimize performance measures indicative of its performance relative to the given task. Interactions between the agent and the environment are very commonly formalized as *Markov Decision Processes* (MDPs).

## 2.1 Markov Decision Processes

We write $\mathscr{P}(\mathcal{X})$ for the set of probability measures on a set $\mathcal{X}$.

**Definition 2.1** (Markov Decision Process (Puterman 2014))**.** A **Markov Decision Process** (MDP) is a tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{P}, \gamma \rangle$ where

- $\mathcal{S}$ is a state space
- $\mathcal{A}$ is an action space
- $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^{\geq 0}$ is a reward function

- $\mathcal{P} : \mathcal{S} \times \mathcal{A} \to \mathscr{P}(\mathcal{S})$ is a probabilistic transition function
- $\gamma \in (0, 1)$ is a discount factor

The semantics of an MDP are to be interpreted as follows: the agent moves around the state space $\mathcal{S}$, which consists of the possible configurations of an environment of interest. The agent interacts with the environment at a particular state $s \in \mathcal{S}$ by choosing an action $a \in \mathcal{A}$, which in turn provides a reward $\mathcal{R}(s, a) \in \mathbb{R}^{\geq 0}$ and (stochastically) transitions the agent to a new state $s' \sim \mathcal{P}(\cdot|s, a)$. In this work we will assume that $\mathcal{S}$ and $\mathcal{A}$ are finite, and that the reward function is bounded by some constant $R_{\text{MAX}}$.

As MDPs are Markovian by construction, an agent need only know their current state (rather than how they got there) in order to decide on a course of action. The strategy of the agent is captured by a *policy* $\pi : \mathcal{S} \to \mathscr{P}(\mathcal{A})$, which (stochastically) chooses actions based on the current state[1]. For a fixed policy $\pi$, the dynamics can also be seen to form a Markov chain with transition kernel $\mathcal{P}^\pi(s'|s) := \sum_a \pi(a|s)\mathcal{P}(s'|s, a)$.

The task of an agent is to pick a sequence of actions so as to maximize its lifetime sum of rewards. The sum of rewards collected by an agent throughout its trajectory is called the *return*. In this thesis we are concerned with *infinite horizon* problems, where the agent is tasked with maximizing rewards over infinite trajectories. In infinite horizon problems, future rewards are exponentially weighted by $\gamma$. The agent, then, seeks to pick a sequence of actions so as to maximize its *expected discounted return*. For each $\pi$ we define a value function $V^\pi : \mathcal{S} \to \mathbb{R}^{\geq 0}$ which assigns to each state the expected discounted return that the agent receives when starting at the state and following the policy $\pi$ and the dynamics of the MDP:

$$V^\pi(s) := \mathbb{E}_{\mathcal{P},\pi} \left[ \sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, a_t) \mid s_0 = s \right],$$

[1] In general the policies could be time-dependent as well, but as we will see it is enough to consider stationary (time-independent) policies.

where $\mathbb{E}_{\mathcal{P},\pi}$ denotes the expectations of random trajectories $(s_0, a_0, s_1, a_1, ...)$ defined inductively by $a_i \sim \pi(\cdot|s_i)$ and $s_{i+1} \sim \mathcal{P}(\cdot|s_i, a_i)$.

The optimal policy $\pi^\star$ is that which maximizes expected discounted returns for every state, i.e. $V^{\pi^\star}(s) \geq V^\pi(s)$ for every $\pi$ and $s \in \mathcal{S}$. In this setting, there always exists an *optimal policy* $\pi^\star$ which is deterministic and stationary (Puterman 2014). The value function for the optimal policy will simply be written as $V^\star := V^{\pi^\star}$.

A closely related object is the *action-value function* of a policy $\pi$, which is defined over state-action pairs. The action-value function $Q^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^{\geq 0}$ is defined as the value of first taking action $a$ and thereafter following policy $\pi$:

$$Q^\pi(s, a) = \mathbb{E}_{\mathcal{P},\pi} \left[ \sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, a_t) \mid s_0 = s, a_0 = a \right].$$

We note that the value function can be recovered from the action-value function (and vice-versa) via the relations

$$V^\pi(s) = \sum_a \pi(a|s) Q^\pi(s, a)$$
$$Q^\pi(s, a) = \mathcal{R}(s, a) + \gamma \sum_{s'} \mathcal{P}(s'|s, a) V^\pi(s').$$

Given the optimal action-value function $Q^\star$, one can recover an optimal policy $\pi^\star$ by simply taking *greedy* actions at every state, i.e.:

$$\pi^\star(s) = \operatorname*{argmax}_{a \in \mathcal{A}} Q^\star(s, a),$$

and thus solving for the optimal policy boils down to finding the optimal value function or action-value function.

Given a sum of discounted rewards observed during a trajectory, we can “chop up“ the time-steps of the trajectory in two components: the immediate reward under policy $\pi$, and the sum of discounted rewards when starting from the next state. Of course, the sum of discounted rewards when starting from the next state is exactly the value function for that state! This leads to a recursive equation for $V^\pi$ – the famous *Bellman equation*:

$$\begin{aligned} V^\pi(s) &= \sum_{a \in \mathcal{A}} \pi(a|s) \left\{ \mathcal{R}(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}(s'|s,a) V^\pi(s') \right\} \\ &= \mathcal{R}^\pi(s) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}^\pi(s'|s) V^\pi(s'), \end{aligned} \tag{2.1}$$

where we introduced the notation $\mathcal{R}^\pi(s) = \mathbb{E}_{a \sim \pi(\cdot|s)}\left[\mathcal{R}(s,a)\right]$. There is a corresponding equation for action-value functions: $Q^\pi(s,a) = \mathcal{R}(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}(s'|s,a) V^\pi(s')$. For the optimal value and optimal action-value functions, there are analogous *Bellman optimality* equations:

$$\begin{aligned} V^\star(s) &= \max_a \left\{ \mathcal{R}(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}(s'|s,a) V^\star(s') \right\} \\ Q^\star(s,a) &= \mathcal{R}(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}(s'|s,a) V^\star(s') \end{aligned} \tag{2.2}$$

The optimality equations can be readily derived by recalling that the optimal policy is greedy with respect to $Q^\star$, and using $V^\star(s) = \max_a Q^\star(s,a)$.

Since $\mathcal{S}$ and $\mathcal{A}$ are finite sets, we can view value functions as vectors over $\mathbb{R}^{|\mathcal{S}|}$. We will write $\mathcal{R}^\pi = \left[\mathcal{R}^\pi(s)\right]_s \in \mathbb{R}^{|\mathcal{S}|}$ as the corresponding vector of rewards and $\mathcal{P}^\pi = \left[\mathcal{P}^\pi(s'|s)\right]_{s,s'} \in \left[\mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}\right]$ for the probability matrix of transitions.[2] In vector form, equation (2.1) becomes

$$V^\pi = \mathcal{R}^\pi + \gamma \mathcal{P}^\pi V^\pi \tag{2.3}$$

This is simply a linear system of equations, which can be solved as $V^\pi = (I_{|\mathcal{S}|} - \gamma \mathcal{P}^\pi)^{-1} \mathcal{R}^\pi$ (the matrix $I_{|\mathcal{S}|} - \gamma \mathcal{P}^\pi$ is invertible since $\mathcal{P}^\pi$ is a stochastic matrix (Puterman 2014, Corollary C.4)).[3] Of course, even if $\mathcal{P}^\pi$ and $\mathcal{R}^\pi$ were known, computing this matrix inverse is

[2] $[A \to B]$ is the set of functions $f : A \to B$.

[3] $I_n$ is the identity matrix on $\mathbb{R}^{n \times n}$

prohibitively expensive in most applications since naïve matrix multiplication is of order $\mathcal{O}(|\mathcal{S}|^3)$. To bypass this issue, we can introduce an operator (suitably called the Bellman operator) which will allow us to iteratively solve the Bellman equations.

## 2.2 Bellman Equations and Dynamic Programming

The Bellman operator $\mathcal{T}^\pi : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}$ for a policy $\pi$ is the affine transformation given by

$$\mathcal{T}^\pi V = \mathcal{R}^\pi + \gamma \mathcal{P}^\pi V.$$

There also exists an analogous *Bellman optimality operator* $\mathcal{T}^\star : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}$ defined by

$$\mathcal{T}^\star V = \max_\pi \mathcal{T}^\pi V.$$

Then, determining the value function for policy $\pi$ or the optimal value function boil down to solving the fixed point equations

$$V^\pi = \mathcal{T}^\pi V^\pi \quad \text{or} \quad V^\star = \mathcal{T}^\star V^\star,$$

respectively. The advantage of this formulation is that one can use tools from *fixed-point* theory. One such tool, particularly prevalent and useful in RL, is the *Banach Fixed Point Theorem* (Banach 1922). A complete metric space $(\mathcal{X}, d)$ is a metric space in which every Cauchy sequence converges to a point in $\mathcal{X}$. A contraction mapping $f : \mathcal{X} \to \mathcal{X}$ satisfies $d(f(x), f(y)) \leq \alpha d(x, y)$ for each $x, y \in \mathcal{X}$ and for some $\alpha < 1$.

**Theorem 2.1** (Banach Fixed Point Theorem). *Let $(\mathcal{X}, d)$ be a complete metric space and $f$ be an $\alpha$-contraction mapping. Then $f$ has a unique fixed point $f(x^\star) = x^\star$.*
*Furthermore, for any $x_0 \in X$, the sequence $f^n(x_0) \xrightarrow{n\to\infty} x^\star$ and*

$$d\left(x^\star, f^n(x_0)\right) \leq \frac{\alpha^n}{1-\alpha} d\left(x_0, f(x_0)\right)$$

It turns out that the Bellman operator and Bellman optimality operator are $\gamma$-contractions with respect to the *max-norm* (i.e. $\ell_\infty$ norm) $\|v\|_\infty = \max_{s \in \mathcal{S}} |v(s)|$.

**Corollary 2.1.1.** *Let $V_0 \in \mathbb{R}^{|\mathcal{S}|}$. Then the following sequences converge uniformly at a geometric rate:*

$$(\mathcal{T}^\pi)^n V_0 \xrightarrow{n \to \infty} V^\pi \tag{PE}$$

$$(\mathcal{T}^\star)^n V_0 \xrightarrow{n \to \infty} V^\star \tag{VI}$$

These two updates are called Policy Evaluation and Value Iteration, respectively. Policy Evaluation repeatedly applies the Bellman operator $\mathcal{T}^\pi$ in order to *evaluate* the value of a candidate policy $\pi$. To find the optimal value function using Policy Evaluation, we perform a *Policy Improvement* step after evaluation, which updates the policy to be greedy with respect to the recently-evaluated value function. Alternating Policy Evaluation and Policy Improvement leads to a monotonic improvement of the policies at every step, from which convergence to the optimal policy follows. On the other hand, Value Iteration directly applies the optimality operator until convergence to $V^\star$ is attained. These two methods belong to the *Dynamic Programming* (DP) family of algorithms (Bellman 1966).

## 2.3 Value-based Reinforcement Learning

In the previous section, we saw DP algorithms which solve for the optimal policy when a full specification of an MDP was given. The main challenges encountered here are computational: the algorithms scale poorly to large state and action spaces (the so-called "curse of dimensionality"(Bellman 1966)). Furthermore, in practice, one is unlikely to be given a full specification of the environment. This is the realm of *reinforcement learning*, which deals with the problem of solving an MDP when the MDP is not given (that is, the transition and reward functions are not known in advance).

There exists a wide class of RL methods, only some of which we can cover here. We will focus on *value-based* methods for solving MDPs, which attempt to iteratively solve or approximate value functions $V^{\pi/\star}$ or action-value function $Q^{\pi/\star}$. To bypass computing the expectation in the Bellman equation, most RL algorithms are *sampling-based*. In other words, value functions must be learned via trial-and-error by collecting trajectories from the environment. All algorithms under consideration will adopt the form (Sutton and Barto 1998):

$$\texttt{NewEstimate} \leftarrow (1 - \texttt{StepSize}) \times \texttt{OldEstimate} + \texttt{StepSize} \times \texttt{Target} \tag{2.4}$$

Algorithms fall in one of two categories: *evaluation* for evaluating the (action-)value function of a policy, or *control* for estimating the optimal (action-)value function. We present the basic tool-kit of value-based methods. We will consider *synchronous* updates, where every state is updated at every iteration.

### Monte Carlo Evaluation

In Monte Carlo algorithms, the targets are given by the discounted sum of rewards collected during a sample trajectory under policy $\pi$. Such a discounted sum of rewards will also be called a *sample return*. The algorithm iteratively updates in the following manner:

$$V_{k+1}(s) = (1 - \alpha_k)V_k(s) + \alpha_k G_k^\pi(s), \tag{MCE}$$

where $G_k^\pi(s)$ is a random discounted return starting at state $s$ and following policy $\pi$. In the case of $\alpha_k = 1/k$, the $k^{\text{th}}$ iteration of a Monte Carlo algorithm is simply the average return of $k$ independent trajectories. As $k \to \infty$, by the strong law of large numbers, these averages will converge to the true value function $V^\pi$ (Bertsekas and Tsitsiklis 1996).

### Temporal-Difference Learning

In the Temporal-Difference family of methods, we *bootstrap* from current estimates rather than collecting complete trajectories. The algorithms evaluate a value function for a given policy $\pi$. The simplest algorithm, TD(0), has the following updates:

$$V_{k+1}(s) = (1-\alpha)V_k(s) + \alpha\left(\mathcal{R}(s,a) + \gamma V_k(s')\right) \qquad \begin{array}{l} a \sim \pi(\cdot|s) \\ s' \sim \mathcal{P}(\cdot|s,a) \end{array} \tag{TD(0)}$$

Interpolating between Monte Carlo methods and TD(0), we have the TD($\lambda$) algorithm. The TD($\lambda$) algorithm, for $\lambda \in [0,1]$ is a weighted average of $n$-step returns:

$$V_{k+1}(s) = (1-\alpha)V_k(s) + \alpha(1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}\underbrace{\Big(\sum_{i=0}^{n}\gamma^i\mathcal{R}(s_i,a_i) + \gamma^n V_k(s_n)\Big)}_{n\text{-step return}}, \tag{TD($\lambda$)}$$

where $(s, a_0, s_1, a_1, ...)$ is a trajectory collected from the environment by following the policy. Taking $\lambda = 1$ recovers the Monte Carlo method.

**SARSA**

For evaluating action-value functions for a policy $\pi$, we have the SARSA algorithm:

$$Q_{k+1}(s,a) = (1-\alpha)Q_k(s,a) + \alpha\left(\mathcal{R}(s,a) + \gamma Q_k(s',a')\right) \qquad \begin{array}{l} s' \sim \mathcal{P}(\cdot|s,a) \\ a' \sim \pi(\cdot|s') \end{array} \tag{SARSA}$$

**Q-Learning**

Finally, for learning the optimal action-value function $Q^\star$, we have the Q-Learning algorithm:

$$Q_{k+1}(s,a) = (1-\alpha)Q_k(s,a) + \alpha\left(\mathcal{R}(s,a) + \gamma\max_{a'} Q_k(s',a')\right) \quad s' \sim \mathcal{P}(\cdot|s,a) \tag{Q-Learning}$$

## 2.4 Optimal Transport and the Coupling Method

We provide some necessary mathematical background on couplings and optimal transport, which will be invaluable in proceeding sections.

**Definition 2.2** (Couplings)**.** Let $\mu \in \mathscr{P}(\mathcal{X}), \nu \in \mathscr{P}(\mathcal{Y})$ be two probability measures over a space $\mathcal{X}$ and $\mathcal{Y}$, respectively. A *coupling* of $\mu$ and $\nu$ is a joint distribution $\lambda \in \mathscr{P}(\mathcal{X} \times \mathcal{Y})$ such that the marginals of $\lambda$ are $\mu$ and $\nu$, respectively. Formally, for every measurable set $\mathcal{A}$ of $\mathcal{X}$ and $\mathcal{B}$ of $\mathcal{Y}$:

$$\lambda(\mathcal{A} \times \mathcal{Y}) = \mu(\mathcal{A}) \quad \& \quad \nu(\mathcal{B}) = \lambda(\mathcal{X} \times \mathcal{B}).$$

For finite spaces this can be rewritten as:

$$\sum_{y \in \mathcal{Y}} \lambda(x, y) = \mu(x) \quad \& \quad \nu(y) = \sum_{x \in \mathcal{X}} \lambda(x, y),$$

for each $x \in \mathcal{X}$, $y \in \mathcal{Y}$. In the language of random variables, a pair $(X, Y)$ is a coupling of $(\mu, \nu)$ if $X \sim \mu$ and $Y \sim \nu$.

Roughly speaking, the intuition is that the random variables $X$ and $Y$ can be correlated, however when we ignore the behaviour of $Y$ (resp. $X$) by integrating over it's behaviour, $X$ (resp. $Y$) must respect the original distribution. The set of couplings for two distributions $(\mu, \nu)$ is denoted by $\Lambda(\mu, \nu)$. The set is always non-empty, and many possible couplings can exist for a given pair of measures. In particular, the *independent* coupling $\lambda(\mathcal{A} \times \mathcal{B}) = \mu(\mathcal{A})\nu(\mathcal{B})$ always exists. As the name suggests, the independent coupling is equivalent to assuming that the two measures are independent: the probability of the pair $(x, x')$ occurring under the coupling is simply the probability of $x$ occurring under $\mu$ times the probability of $x'$ occurring under $\nu$. On the other end of the spectrum, the *diagonal* coupling $\lambda(x, x') = \mathbb{1}_{[x=x']}\mu(x) = \mathbb{1}_{[x=x']}\nu(x')$ is the "strongest" possible coupling: $\mu(\cdot)$ samples $x$ if and only if $\nu(\cdot)$ does, and vice-versa![4]

Another important coupling is the *optimal coupling*. Supposing we had a cost function $d(x, y)$ on $\mathcal{X} \times \mathcal{X}$ (for our purposes we will always think of cost functions simply as metrics

[4] $\mathbb{1}_A(x)$ denotes the indicator function of a set $A$

on $\mathcal{X}$), the optimal coupling is the one which minimizes the transport problem

$$\inf_{\lambda \in \Lambda(\mu,\nu)} \int_{\mathcal{X}\times\mathcal{X}} d(x,x')\lambda(\mathrm{d}x\,,\mathrm{d}x') = \inf_{(X,Y)\in\Lambda(\mu,\nu)} \mathbb{E}\, d(X,Y)$$

When the cost function is a metric, the transport problem above is a metric in its own right, which is called the *Wasserstein metric*.[5] A Wasserstein distance can be defined for any base metric, thus we define the functional $\mathcal{W}(d)(\mu,\nu) \coloneqq \inf_{\lambda\in\Lambda(\mu,\nu)} \int_{\mathcal{X}\times\mathcal{X}} d(x,x')\lambda(\mathrm{d}x\,,\mathrm{d}x')$ which maps a base metric to its associated optimal transport problem. The functional maps a metric between elements of $\mathcal{X}$ to a metric between *distributions* on elements of $\mathcal{X}$ – we say that it *lifts* the metric. We will revisit the concept of liftings in Chapter 3. More generally, for any $p \in [1,\infty)$, we can consider the $p$-Wasserstein metric $\mathcal{W}_p(d)(\mu,\nu) = \inf_\lambda \left(\int d(x,x')^p\lambda(\mathrm{d}x\,,\mathrm{d}x')\right)^{1/p}$. So as to make the metric take finite values, we restrict the measures under consideration to the set $\mathscr{P}_p(\mathcal{X}) \coloneqq \left\{\mu\in\mathscr{P}(\mathcal{X}) : \int d(x_0,x)^p\mu(\mathrm{d}x) < +\infty\right\}$, where the choice of $x_0$ is arbitrary.

In one sentence, the *coupling method* is a proof strategy for bounding the distance between two probability distributions by choosing a coupling which guarantees that pairs of elements drawn from the coupling will be equal with high probability. The freedom to choose any coupling comes from the definition of the Wasserstein distances as minimization problems, letting $(X,Y)$ by *any* coupling of $(\mu,\nu)$ we have:

$$\mathcal{W}(d)(\mu,\nu) \le \mathbb{E}[d(X,Y)]$$

A particular case is the well-known *total variation* inequality (Lindvall 2002, Chapter 1.2). The total variation metric $d_{\mathrm{TV}}(\mu,\nu)$, be recovered (up-to a factor of $2$) by taking the trivial metric $d(x,y) = \mathbb{1}_{[x=y]}$. Then:

$$d_{\mathrm{TV}}(\mu,\nu) \le 2\mathbb{P}[X \ne Y]$$

[5] We use the term Wasserstein metric so as to be in accordance with the literature, although that name is not historically accurate: see the discussion in (Villani 2008, p.86)

Thus, choosing any coupling which equates the two processes with high probability guarantees convergence as well.

While we are here, we will state two properties of the Wasserstein distances which will be needed in Chapter 5. The first one is that an optimal coupling (i.e. one minimizing the transport cost) always exists. A Polish metric space is a complete separable metric space. The vector space $\mathbb{R}^n$ is the only example of a Polish space which we will need to consider.

**Theorem 2.2** (Existence of optimal coupling (Villani 2008, Theorem 4.1))**.** *Let $(\mathcal{X}, d)$ be a Polish space, and $\mu, \nu \in \mathscr{P}(\mathcal{X})$. Then there exists an optimal coupling of $(\mu, \nu)$ which minimizes the transport problem. That is, there exists $(X^\star, Y^\star)$ such that*

$$\mathbb{E}\, d(X^\star, Y^\star) = \mathcal{W}(d)(\mu, \nu) = \inf_{(X,Y)} \mathbb{E}\, d(X, Y)$$

The second one is that the Wasserstein lifting preserves Polishness.

**Theorem 2.3** (Completeness of $\mathscr{P}_p(\mathcal{X})$ (Villani 2008, Theorem 6.16))**.** *Let $(\mathcal{X}, d)$ be a Polish metric space, and equip $\mathscr{P}_p(\mathcal{X})$ with the $\mathcal{W}_p(d)$ metric. Then $\mathscr{P}_p(\mathcal{X})$ is a Polish metric space. In particular, it is complete.*

## 2.5 State Abstractions and Bisimulation Relations

In many practical applications, the state space of the MDP is simply too large to allow one to compute the value functions exactly without the use of *state abstraction*. State abstraction refers to finding a *representation* of smaller dimension which will allow tractable computation of value functions. In this thesis, we will be concerned with abstraction via *state aggregation*, which consists of aggregating states into clusters. The set of aggregated states forms the *abstract MDP*.

One notion used to capture behavioral similarity of states is called bisimulation. Bisimulation has long been used in the fields of concurrency and formal verification for provably verifying the correctness and safety of processes and systems (Milner 1980; Park 1981). Variants have been proposed for different notions of transition systems, including a probabilistic version for probabilistic transition systems by (Larsen and Skou 1991). An extension to MDPs was proposed by (Givan, Dean, and Greig 2003). Bisimulation is a canonical equivalence for analyzing the behaviour of transition systems and clustering equivalent states in overly large systems – the optimal policy and optimal value functions will be preserved in the abstract MDP (Li, Walsh, and Littman 2006).
A binary relation on $\mathcal{S}$ is simply a subset $\Phi \subseteq \mathcal{S} \times \mathcal{S}$. We write $s\Phi t$ if $(s,t) \in \Phi$. Furthermore, a binary relation is an *equivalence relation* if, for each $s,t,w \in \mathcal{S}$: (1) $s\Phi s$, (2) $s\Phi t \Rightarrow t\Phi s$, and (3) $s\Phi t$ and $t\Phi w \Rightarrow s\Phi w$. An *equivalence class* of an equivalence relation is a set of the form $[s] \coloneqq \{s' \in \mathcal{S} \mid s\Phi s'\}$. Equivalence classes fully partition the original state space, we write $\mathcal{S}/\Phi$ for the set of these classes.

**Definition 2.3** (Bisimulation)**.** A *bisimulation relation* $\mathcal{U}$ on $\mathcal{S}$ is an equivalence relation such that $s\mathcal{U}s'$ implies:

(1) $\forall a \in \mathcal{A}, \mathcal{R}(s,a) = \mathcal{R}(s',a)$ and

(2) $\forall a \in \mathcal{A}, \forall \mathcal{C} \in \mathcal{S}/\mathcal{U}, \mathcal{P}(\mathcal{C}|s,a) = \mathcal{P}(\mathcal{C}|s',a)$,

where $\mathcal{P}(\mathcal{C}|s,a) = \sum_{s'\in\mathcal{C}} \mathcal{P}(s'|s,a)$. We say that $s$ and $s'$ are *bisimilar* and write $s \sim s'$ if there is some bisimulation relation $\mathcal{U}$ relating them.

The picture is as follows: imagine that one is provided with an equivalence relation $\mathcal{U}$. Partition the state space based on the equivalence classes of $\mathcal{U}$ (i.e. cluster together all the states which are related to each other). Then, one has to check that each state in each cluster has the same rewards as every other state in that cluster, and furthermore that the probability of transitioning to other clusters is the same for each state. Thus bisimilar states will have equal rewards and thereafter transition with equal probability to more bisimilar states. Unfortunately, bisimulation is too stringent: any $\varepsilon$ difference in rewards or transition distributions will cause states to break their bisimilarity. To remedy this, quantitative analogues of bisimulation have been proposed.

## 2.6 Bisimulation metrics

The metric analogue of bisimulation was defined by (Desharnais et al. 1999) in the setting of labelled Markov processes (a model similar to MDPs but without rewards). The extension to MDPs was given in (Ferns, Panangaden, and Precup 2004). These metrics allow one to measure "how bisimilar" two states are.

Technically, the bisimulation metrics are in fact *pseudometrics*. A pseudometric $d_{\text{pseudo}}$ obeys the usual axioms of a metric except that the distance between two different points is allowed to be $0$. The *kernel* of a pseudometric is the set of tuples which are assigned a distance of $0$: $\mathrm{Ker}(d_{\text{pseudo}}) = \big\{(s,s') \mid d_{\text{pseudo}}(s,s') = 0\big\}$. Evidently, the kernel of a proper metric is the diagonal set $\Delta = \{(s,s) \mid s \in \mathcal{S}\}$. Much like value functions, bisimulation metrics are defined in terms of a fixed point equation (cf. Equation (2.1)). We will occasionally abuse terminology and say "metric" in lieu of "pseudometric".

**Definition 2.4** (Bisimulation metrics)**.** Define

$$\mathcal{F}(d)(s,s') = \max_a\{(1-\gamma)|\mathcal{R}(s,a) - \mathcal{R}(s',a)| + \gamma\mathcal{W}(d)(\mathcal{P}(\cdot|s,a), \mathcal{P}(\cdot|s',a)\}.$$

Note that $\mathcal{F}$ is an operator that takes a metric and outputs a metric. Then the bisimulation distance $d_\sim$ is defined as the fixed point of $\mathcal{F}$:

$$\begin{aligned} d_\sim(s,s') &= \max_a \{(1-\gamma)|\mathcal{R}(s,a)-\mathcal{R}(s',a)| + \gamma\mathcal{W}(d_\sim)(\mathcal{P}(\cdot|s,a),\mathcal{P}(\cdot|s',a)\} \\ &= \mathcal{F}(d_\sim)(s,s') \quad \forall\, s,s' \in \mathcal{S} \end{aligned}$$

Roughly speaking, the bisimulation metric is defined by measuring the difference in rewards at the current step and the difference in bisimulation distance between the distributions of next states. Similarly to the optimal value function, the bisimulation metric can be obtained as a sequence of iterates, setting $d_0(s,s') = 0$ and $d_{n+1}(s,s') = \max_a\{(1-\gamma)|\mathcal{R}(s,a)-\mathcal{R}(s',a)|+\gamma\mathcal{W}(d_n)(\mathcal{P}(\cdot|s,a),\mathcal{P}(\cdot|s',a))\}$.[6] The bisimulation metrics have many pleasing theoretical properties: they assign a distance of zero to states if and only if they are bisimilar (i.e. $\mathrm{Ker}(d_\sim) = \sim$). Furthermore, the metrics can be used for state abstraction (by aggregating states that are $\varepsilon$ away), and doing so provides one with a formal bound on the approximation error incurred (that is, the difference between the true optimal value function and the approximated one). Unfortunately, they are too impractical to compute for many applications. Each iteration of the recursion requires one to compute a linear program between the transition distributions for every pair of states, which is not only computationally expensive but also requires knowledge of the full model of the MDP. In the next chapter, we investigate alternative metrics for behavioural equivalence, with the goal of maintaining the useful theoretical guarantees of bisimulation metrics while reducing the computational burden and the need for knowledge of the model.

[6] In fact, a bisimulation metric can be seen as the optimal value function of a coupling of two copies of the original MDP (Ferns and Precup 2014).

# Chapter 3

# Temporally Extended Metrics: State Abstraction via Couplings

As mentioned in Section 2.6, the bisimulation metrics have many pleasing properties but are hindered by their computational requirements: namely, full knowledge of the MDP and the need to solve a linear program (via the Wasserstein metric) at every step of the recursion. In this chapter, we investigate alternative metrics for behavioural equivalence based on couplings.

More concretely, our contributions are two-fold: first, we propose a coupling-based generalization of bisimulation which allows for greater flexibility in the comparisons between states (instead of strict reward matching), and consequently in the properties being checked. The generalization builds on couplings, and more specifically on the probabilistic *liftings* of a relation. By considering an arbitrary relation between states $\Phi$ as a *base relation*, we describe how to "lift" this relation to a generalized $\Phi$-bisimulation relation. Second, we consider the class of *quantitative bisimulations* and show how this defines a notion of *temporally extended (TE) metrics*. Intuitively, these metrics compute the minimum value of a chosen base metric for which the states $s$ and $s'$ can remain in that range throughout their dynamics. The TE metrics assign distance 0 to states if and only if they are bisimilar, much like the bisimulation metrics previously defined. However, both the construction and the resulting metric are quite different.

The rest of the chapter is organized as follows. In the next section we introduce some mathematical tools. In Section 3.1 and Section 3.2 we characterize bisimulation via couplings and define the extension of this characterization to arbitrary relations. In Section 3.3 we define the temporally extended metrics. Section 3.4 compares the two metrics by providing some bounds relating them and analyzing the couplings induced by the two metrics. Lastly, we wrap up with a discussion on the benefits and disadvantages of these metrics, highlighting directions for future work.

## 3.1 Bisimulation via Liftings

As seen in section 2.4, one can always construct at least one coupling between two distributions (namely, the independent coupling). Thus, the sheer existence of some coupling does not provide any information. To gain some understanding of the two distributions being coupled, we can require that the *support* of a coupling satisfies some property. This is the notion of the *probabilistic lifting* of a relation (henceforth simply "lifting").

Given a binary relation $\Phi$ between states, the *lifting* $(\Phi)^{\#}$ of that relation allows one to naturally extend $\Phi$ to a relation between *distributions* on states. We write $\mathrm{supp}(\lambda) = \{(s, s') \in \mathcal{S} \times \mathcal{S} \mid \lambda(s, s') > 0\}$ for the support of a distribution.

**Definition 3.1** (Liftings). A coupling $\lambda$ of distributions $(\mu, \nu)$ is a *lifting* of a binary relation $\Phi$ if:

$$\lambda(s, s') > 0 \Rightarrow s\Phi s', \text{ i.e. } \mathrm{supp}(\lambda) \subseteq \Phi.$$

When there exists a lifting of $\mu$ and $\nu$ as above we write $\mu(\Phi)^{\#}\nu$. The lifted relation $(\Phi)^{\#}$ is a new binary relation between distributions.

For example, the diagonal coupling $\lambda(s,s') = \mathbb{1}_{[s=s']}\mu(s)$ is a lifting of the equality relation $(=)$. Since the diagonal coupling is only a valid coupling if $\mu = \nu$, we see that $\mu(=)^{\sharp}\nu \iff \mu = \nu$. As another example, the independent coupling relates every element of $\mathcal{S}$, i.e. lifts the trivial relation $\top = \mathcal{S} \times \mathcal{S}$. We remark on the repeated use of the word lifting for different but related contexts: the lifting of a relation should not be confused with the lifting of a metric via the Wasserstein metric (Section 2.4). Both procedures involve a extending base object (namely, a binary relation or a metric) between states to an object of the same type between distributions on states. For the rest of this chapter the term lifting will be reserved for relations.

In bisimulation, the rewards match at the first step and thereafter the states transition such that the bisimulation relation is preserved. This definition can also be captured in terms of couplings: our first result is that bisimulation is equivalent to the existence of a particular lifting of the states.

**Theorem 3.1.** *A relation $\mathcal{U}$ is a bisimulation relation if and only if $s\mathcal{U}s'$ implies:*

*(1)* $\forall a\, \mathcal{R}(s,a) = \mathcal{R}(s',a)$

*(2)* $\forall a\, \mathcal{P}(\cdot|s,a)\ (\mathcal{U})^{\#}\ \mathcal{P}(\cdot|s',a)$

*Proof.* We prove the forward implication first. Let $\mathcal{U}$ be a bisimulation relation and $s\mathcal{U}t$. The first condition is immediate, since $\mathcal{R}(s,a) = \mathcal{R}(t,a)\ \forall a$ by definition of bisimulation. For the second condition, for each $a$ we pick the couplings

$$\lambda_{a,s,t}(s',t') = \begin{cases} \mathcal{P}(s'|s,a)\mathcal{P}(t'|t,a)/\mathcal{P}([s']|s,a), & s'\mathcal{U}t' \\ 0, & s' \not{\mathcal{U}}\, t' \end{cases}$$

where $[s'] := \{t' \mid s'\mathcal{U}t'\}$ is the equivalence class of $s'$. We note that the coupling depends on each of $a$, $s$, and $t$ although the subscripts may be omitted from now on. The marginals match since:

$$\begin{aligned}\lambda(s', \mathcal{S}) &= \sum_{t' : s'\mathcal{U}t'} \mathcal{P}(s'|s,a)\mathcal{P}(t'|t,a)/\mathcal{P}([s']|s,a) \\ &= \mathcal{P}(s'|s,a) \sum_{t' : s'\mathcal{U}t'} \mathcal{P}(t'|t,a)/\mathcal{P}([s']|s,a) \\ &= \mathcal{P}(s'|s,a)\mathcal{P}([s']|t,a)/\mathcal{P}([s']|s,a) \\ &= \mathcal{P}(s'|s,a),\end{aligned}$$

as $\mathcal{P}([s']|s,a) = \mathcal{P}([s']|t,a)$ by the second condition of bisimulation. Similarly, $\lambda(\mathcal{S}, t') = \mathcal{P}(t'|t,a)$. To make $\lambda$ a lifting of $\mathcal{U}$ we still need to check that $\operatorname{supp}(\lambda) \subseteq \mathcal{U}$, which is evident since $\lambda(s', t')$ is only non-zero when $s'\mathcal{U}t'$.
For the converse, let $\mathcal{U}$ be such that $s\mathcal{U}t$ satisfies conditions 1 and 2. We show that $\mathcal{U}$ is a bisimulation relation. Again, condition 1 is immediate since $\mathcal{R}(s,a) = \mathcal{R}(t,a)\ \forall a$. Next we show that $\forall C \in \mathcal{S}/\mathcal{U}, \mathcal{P}(C|s,a) = \mathcal{P}(C|t,a)$. Let $\lambda$ be the lifting given by condition 2. Note that $\lambda(C,C) = \lambda(C,\mathcal{S})$, since otherwise we could pick $s' \not\mathcal{U}\, t'$ and obtain $(s',t') \in \operatorname{supp}(\lambda)$, which contradicts $\operatorname{supp}(\lambda) \subseteq \mathcal{U}$. Similarly, $\lambda(\mathcal{S}, C) = \lambda(C,C)$. Thus,

$$\mathcal{P}(C|s,a) = \lambda(C,\mathcal{S}) = \lambda(C,C) = \lambda(\mathcal{S},C) = \mathcal{P}(C|t,a)$$

Since $\mathcal{U}$ is a bisimulation relation and $s\mathcal{U}t$, then $s \sim t$. □

The backward implication can also be derived from the remarkable Strassen's theorem on couplings (see e.g. (Lindvall 1999)), which implies that for any $\Phi$, $\mu(\Phi)^{\#}\nu \iff \forall A \subseteq \mathcal{S},\ \mu(A) \leq \nu(\Phi(A))$. Applied to an equivalence class $C$ and using that $\mathcal{U}$ is symmetric gives the bisimulation property.

### A simple example: bisimulation and liftings

As an example, we can consider the bisimilar states in Figure 3.1. To see that $s_0$ and $t_0$ are bisimilar, take the equivalence classes $\mathcal{S}/\mathcal{U} = \{\{s_0, t_0\}, \{s_1, t_1\}, \{s_2, t_2, t_3\}\}$. All states in each class receive the same rewards and transition with equal probability to all other classes, so $\mathcal{U}$ is indeed a bisimulation relation. Now consider the coupling $\lambda$ of $(\mathcal{P}(\cdot|s_0), \mathcal{P}(\cdot|t_0))$ given by the dashed arrows, i.e. $\lambda(s_1, t_1) = \lambda(s_2, t_2) = \lambda(s_2, t_3) = \frac{1}{3}$. This coupling is a *lifting* of $\mathcal{U}$, since all supported states are $\mathcal{U}$-related. The support condition can be interpreted as finding a coupling which only transports mass (via the dashed arrows) to and from states that are bisimilar to each other. Not all couplings are liftings: for instance the trivial coupling $\omega(s_i, t_j) = \mathcal{P}(s_i|s_0)\mathcal{P}(t_j|t_0)$ is not a lifting of $\mathcal{U}$.

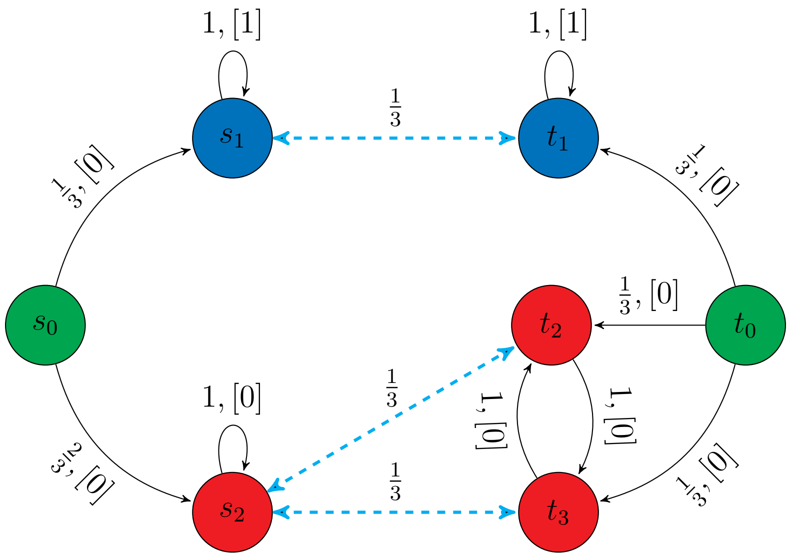

Figure 3.1: Different colours represent different equivalence classes of $\sim$. Rewards are indicated in square brackets, and transition probabilities of the MDP are given before the reward. The dashed blue arrows are not part of the MDP, but rather give the weights of the coupling of $\mathcal{P}(\cdot|s_0)$ and $\mathcal{P}(\cdot|t_0)$.

## 3.2 $\Phi$-bisimulation

Building on the previous result, one can readily generalize the first condition by using a generic relation between states instead of demanding that the rewards be equal.

**Definition 3.2** ($\Phi$-bisimulation)**.** Given a *base* relation $\Phi \subseteq \mathcal{S} \times \mathcal{S}$, a $\Phi$*-bisimulation relation* $\mathcal{U} \subseteq \mathcal{S}\times\mathcal{S}$ is a new relation where the states are $\Phi$-related and their transition distributions are $\mathcal{U}$-lifted. Formally, $s\mathcal{U}s'$ implies:

(1) $s\Phi s'$

(2) $\forall a \; P(\cdot|s,a) \; (\mathcal{U})^{\#} \; \mathcal{P}(\cdot|s',a)$

We define $\Phi$*-bisimulation* $\overset{\Phi}{\sim}$ to be the largest $\Phi$-bisimulation relation.

This allows one to define arbitrary properties that are preserved by the dynamics of the MDP in a systematic way. We remark, firstly, that the second condition is much stronger than merely requiring that the lifting is a coupling supported by $\Phi$, that is, $P(\cdot|s,a) \; (\Phi)^{\#} \; \mathcal{P}(\cdot|s',a)$. Requiring $\mathcal{U}$ to be lifted also demands (in a corecursive manner) that the successor states after a transition are $\Phi$-related *and can themselves exhibit an appropriate coupling*. Secondly, we note that $\overset{\Phi}{\sim}$ is unique and well-defined, since the union of $\Phi$-bisimulation relations is itself a $\Phi$-bisimulation relation. The well-behavedness of $\Phi$-bisimulations depends on their base relations, i.e. $\overset{\Phi}{\sim}$ is reflexive, symmetric, and transitive whenever $\Phi$ has the same property (see Lemma 3.1). Evidently, taking the relation $\Phi = \{(s,t) \mid \mathcal{R}(s,a) = \mathcal{R}(t,a) \; \forall a\}$ gives the usual bisimulation relation which we introduced in Section 2.5 (i.e. $\overset{\Phi}{\sim} = \sim$). However, we can now consider arbitrary comparisons between states (e.g equality of value functions or approximate matching of rewards) as our base relation, and the coupling condition can be used to verify that that property holds at future time steps as well.

A useful result which we will need later is that if the base relation is an equivalence relation, then so is the induced bisimulation. For each property, one has to exhibit an appropriate coupling.

**Lemma 3.1.** *A $\Phi$-bisimulation $\overset{\Phi}{\sim}$ is reflexive, symmetric, and transitive whenever $\Phi$ has the same property.*

*Proof.* Suppose $\Phi$ is reflexive. We find a $\Phi$-bisimulation relation $\mathcal{U}$ s.t. $s\mathcal{U}s$. Pick $\mathcal{U} = \{(s,s)|s \in S\}$. The first condition ($s\Phi s$) is satisfied, by reflexivity of $\Phi$. For the second condition, we pick the "diagonal" coupling $\lambda_a(s', s'') = \mathcal{P}(s'|s,a)$ if $s' = s''$ and $\lambda_a(s', s'') = 0$ otherwise. Evidently the marginals match. The support is contained in $\mathcal{U}$ since $\forall a, \mathrm{supp}(\lambda_a) = \mathcal{U}$.
Suppose $\Phi$ is symmetric, and that $s \overset{\Phi}{\sim} t$. Since there exists a $\Phi$-bisimulation relation $\mathcal{U}$ s.t. $s\mathcal{U}t$, we define $\bar{\mathcal{U}} = \{(t,s)|(s,t) \in \mathcal{U}\}$. We show $\bar{\mathcal{U}}$ is a $\Phi$-bisimulation relation. For the first condition, $t\bar{\mathcal{U}}s \Rightarrow s\mathcal{U}t \Rightarrow s\Phi t \Rightarrow t\Phi s$ since $\Phi$ is symmetric. For the second, we pick the mirror coupling $\psi_a(t', s') = \lambda_a(s', t')$. Then, $(t', s') \in \mathrm{supp}(\psi_a) \Rightarrow (s', t') \in \mathrm{supp}(\lambda_a) \Rightarrow (s', t') \in \mathcal{U} \Rightarrow (t', s') \in \bar{\mathcal{U}}$. Thus $t \overset{\Phi}{\sim} s$.
Finally suppose $\Phi$ is transitive, and $s \overset{\Phi}{\sim} w$ and $w \overset{\Phi}{\sim} t$. Let $\mathcal{U}_1$ and $\mathcal{U}_2$ be the respective $\Phi$-bisimulation relations. We show that $\mathcal{U}_1 \circ \mathcal{U}_2 \subseteq \overset{\Phi}{\sim}$, where $\mathcal{U}_1 \circ \mathcal{U}_2 = \{(s,t) \mid \exists w : (s,w) \in U_1 \;\&\; (w,t) \in U_2\}$. The first condition ($s\Phi t$) is met since $\Phi$ is transitive. For the second condition, we let $\lambda_{a,1}$ and $\lambda_{a,2}$ be the liftings for $\mathcal{P}(\cdot|s,a)(\mathcal{U}_1)^{\#}\mathcal{P}(\cdot|w,a)$ and $\mathcal{P}(\cdot|w,a)(\mathcal{U}_2)^{\#}\mathcal{P}(\cdot|t,a)$, respectively. We pick the 'transitive' coupling $\lambda_a(s', t') = \sum_{w' \in \mathrm{supp}(\mathcal{P}(\cdot|w,a))} \frac{\lambda_{a,1}(s',w')\lambda_{a,2}(w',t')}{\mathcal{P}(w'|w,a)}$. One can check that the marginals match. For the support condition, $(s', t') \in \mathrm{supp}(\lambda_a) \Rightarrow \exists w'$ s.t. $\lambda_{a,1}(s', w') > 0 \;\&\; \lambda_{a,2}(w', t') > 0 \Rightarrow (s', w') \in \mathrm{supp}(\lambda_{a,1}) \;\&\; (w', t') \in \mathrm{supp}(\lambda_{a,2}) \Rightarrow (s', w') \in \mathcal{U}_1 \;\&\; (w', t') \in \mathcal{U}_2 \Rightarrow (s', t') \in \mathcal{U}_1 \circ \mathcal{U}_2$. □

## 3.3 Temporally Extended Metrics

Although the framework presented in Section 3.1 is agnostic with respect to the base relation $\Phi$, we will be focusing on the setting of *quantitative relations*. These are relations parametrized by the use of a real number $\varepsilon \geq 0$, which arise from a *base pseudometric* $\delta : S \times S \to \mathbb{R}^{\geq 0}$ on states (or state-action pairs). More formally, given a base pseudometric $\delta : S \times S \to \mathbb{R}^{\geq 0}$, a quantitative relation $\delta_\varepsilon$ is the relation consisting of pairs that are $\varepsilon$ away in the metric: $\delta_\varepsilon := \delta^{-1}\left([0, \varepsilon]\right) = \{(s, s') \mid \delta(s, s') \leq \varepsilon\}$. We note the distinction between the metric $\delta$ and the relations $\delta_\varepsilon$ derived from the metric. We call a bisimulation arising from such a quantitative relation a *quantitative bisimulation*, and will write $\overset{\delta}{\sim}_\varepsilon$ rather than $\overset{\delta_\varepsilon}{\sim}$. An example of quantitative relations is *approximate* reward equality defined by $s \rho_\varepsilon s'$ if $\left[\max_a |\mathcal{R}(s, a) - \mathcal{R}(s', a)| \leq \varepsilon\right]$, derived from the base metric $\rho(s, s') = \max_a |\mathcal{R}(s, a) - \mathcal{R}(s', a)|$.

In the context of quantitative bisimulations, we can define the new metric by taking the infimum over the $\varepsilon$ parameter. We call these the *temporally extended (TE) metrics*. The TE metric finds the minimum $\varepsilon$ such that the states are $\delta_\varepsilon$-bisimilar. That is, the two states are a distance of $\varepsilon$ away (in the base metric $\delta$) and can be coupled corecursively so that future states are $\varepsilon$ away and can themselves be coupled. A temporal extension can be defined for any base pseudometric.

**Definition 3.3** (TE metric)**.** Given a base metric $\delta$ and a corresponding collection of quantitative relations $\{\delta_\varepsilon\}_{\varepsilon \geq 0}$, the *TE metric* for $\delta$ is defined by

$$d_\tau(\delta)(s, s') = \inf\left\{\varepsilon \mid s \overset{\delta}{\sim}_\varepsilon s'\right\}.$$

This construction does indeed give well-defined pseudometrics. The proof follows from the symmetry, transitivity, and additivity of the relations $\overset{\delta}{\sim}_\varepsilon$ , which we derive first.

**Lemma 3.2.** *For all $\varepsilon_1, \varepsilon_2 > 0$ and $s, t, w \in \mathcal{S}$, quantitative bisimulations are*

- *reflexive:* $s \overset{\Phi}{\sim}_{\varepsilon} s$,
- *symmetric:* $s \overset{\Phi}{\sim}_{\varepsilon} t \Rightarrow t \overset{\Phi}{\sim}_{\varepsilon} s$,
- *and additive:* $s \overset{\delta}{\sim}_{\varepsilon_1} t \,\&\, t \overset{\delta}{\sim}_{\varepsilon_2} w \Rightarrow s \overset{\delta}{\sim}_{\varepsilon_1+\varepsilon_2} w$.

*Proof.* Items 1 and 2 follow from Lemma 3.1: $\delta_\varepsilon$ is reflexive and symmetric since $\delta(s,s) = 0$ and $\delta(s,t) = \delta(t,s)$ by definition of pseudometrics, and therefore $\overset{\Phi}{\sim}_{\varepsilon}$ is as well. Item 3 follows from the triangle inequality (since $\delta(s,w) \leq \varepsilon + \varepsilon'$), choosing the transitive coupling from the proof of Lemma 3.1 as a lifting of $s \overset{\Phi}{\sim}_{\varepsilon+\varepsilon'} w$. □

**Theorem 3.2.** *Given a base pseudometric $\delta$, the TE metric $d_\tau(\delta)$ is indeed a pseudometric on $\mathcal{S}$.*

*Proof.* We check the axioms, writing $d_\tau(s,s')$ instead of $d_\tau(\delta)(s,s')$:
1. Note that $s \overset{\Phi}{\sim}_0 s$ by Lemma 3.2, thus $d_\tau(s,s) = \inf_{\varepsilon\geq 0}\{s \overset{\Phi}{\sim}_{\varepsilon} s\} = 0$.
2. Note that $s \overset{\Phi}{\sim}_{\varepsilon} t \Rightarrow t \overset{\Phi}{\sim}_{\varepsilon} s$, thus $d_\tau(s,t) = \inf_\varepsilon\{s \overset{\Phi}{\sim}_{\varepsilon} t\} = \inf_\varepsilon\{t \overset{\Phi}{\sim}_{\varepsilon} s\} = d_\tau(t,s)$.
3. Let $A = A_1 + A_2 = \{\varepsilon_1 + \varepsilon_2 | s \overset{\Phi}{\sim}_{\varepsilon_1} w \,\&\, w \overset{\Phi}{\sim}_{\varepsilon_2} t\}$, and $B = \{\varepsilon | s \overset{\Phi}{\sim}_{\varepsilon} t\}$. Note $A \subseteq B$ since $s \overset{\Phi}{\sim}_{\varepsilon_1+\varepsilon_2} t$ by Proposition 3.2. Thus $d_\tau(s,t) = \inf(B) \leq \inf(A) = \inf(A_1) + \inf(A_2) = d_\tau(s,w) + d_\tau(w,t)$. □

Moreover, the temporally extended metrics assign distance 0 to states if and only if they are perfectly bisimilar in the base metric (i.e. they are $\delta_0$-bisimilar). For the reward metric $\rho(s,s') = \max_a |\mathcal{R}(s,a) - \mathcal{R}(s',a)|$, this implies:

**Theorem 3.3.** *Classical bisimulation corresponds exactly to the kernel of the temporal extension of the reward metric $\rho$, i.e.*

$$s \sim s' \iff d_\tau(\rho)(s,s') = 0.$$

For reward differences, the bisimulation metrics share this property, although our metrics are more general. Furthermore, despite the kernels matching, the TE metrics are *not the same* as the bisimulation metrics, both in construction and in the distances they assign.

### A simple example revisited

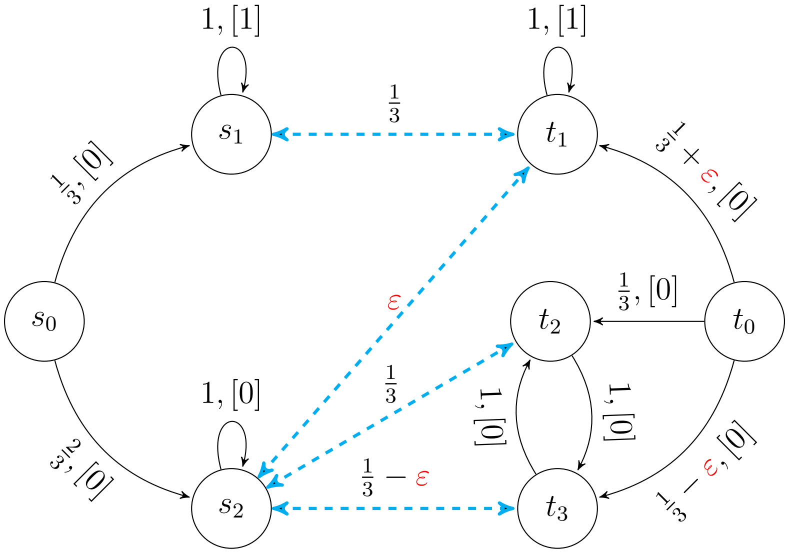

Figure 3.2: Rewards indicated in square brackets. Dashed arrows give the weights of the coupling of $p(s_0)$ and $p(t_0)$.

We consider the almost-bisimilar states in Figure 3.2, examined with $\rho$ as the base metric. In this example, all states are $\rho_1$-bisimilar, but not $\rho_0$-bisimilar. This is captured by the metric: one needs to couple $(s_2, t_1)$, since the marginal onto $t_1$ has to equal $1/3 + \varepsilon$ and $s_1$ only has $1/3$ to spare. Since $(s_2, t_1) \in \text{supp}(\lambda)$ and $|\mathcal{R}(s_2) - \mathcal{R}(t_1)| = 1$, then $d_\tau(\rho)(s_0, t_0) = 1$. This example highlights the discontinuous behaviour of the TE metric – in this case the $\varepsilon$ change in the transition distributions of the MDP means that the states can no longer be matched exactly and we must therefore couple states which have a reward difference of 1. We discuss a possible fix for this discontinuity in Section 3.6.

## 3.4 Comparing Bisimulation Metrics and Temporally Extended Metrics

In this section, we compare the temporally extended metrics with the bisimulation metrics. Results in this section are given in terms of reward metric $\rho$ but can be generalized to arbitrary base metrics. The proofs elucidate the very useful properties of liftings.

### 3.4.1 Bounds

Our first result relates the TE metric and the bisimulation metric with an upper bound.

**Theorem 3.4.** *The temporal extension of $\rho$ upper bounds the bisimulation metric: $\forall s, s' \in \mathcal{S}$,*

$$d_{\sim}(s, s') \leq d_{\tau}(\rho)(s, s').$$

*Proof.* Let $d_n$ denote the $n^{\text{th}}$ iteration of the recursion $d_n = \mathcal{F}^n(d_0)$, with $d_0$ being the zero metric. We will simply write $d_\tau(s, s')$ instead of $d_\tau(\rho)(s, s')$. We proceed by induction, showing that $\forall s, t$, $d_n(s, s') \leq (1-\gamma)\sum_{i=0}^{n} \gamma^i d_\tau(s, s')$. The base case is

$$d_1^\delta(s, s') = \max_a \{(1-\gamma)|\mathcal{R}(s,a) - \mathcal{R}(s',a)|\} \leq (1-\gamma) d_\tau(s, s'),$$

using $\max_a |\mathcal{R}(s,a) - \mathcal{R}(s',a)| \leq d_\tau(s, s')$. For the induction step we upper bound the min-cost coupling of the Wasserstein metric problem with the liftings $\lambda_a \in$

$\Lambda\left(\mathcal{P}(\cdot|s,a),\mathcal{P}(\cdot|s',a)\right)$ given by $\overset{\rho}{\sim}_{d_\tau(s,s')}$.

$$\begin{aligned}
d_{n+1}(s,s') &= \max_a \left\{(1-\gamma)|\mathcal{R}(s,a)-\mathcal{R}(s',a)| + \gamma\mathcal{W}(d_n)\left(\mathcal{P}(\cdot|s,a),\mathcal{P}(\cdot|s',a)\right)\right\} \\
&\leq (1-\gamma)d_\tau(s,s') + \gamma\sum_{k,j}\lambda_a(s_k,s_j)d_n(s_k,s_j) && (\lambda_a \text{ is a coupling}) \\
&\leq (1-\gamma)d_\tau(s,s') \\
&\quad + \gamma\sum_{k,j}\lambda_a(s_k,s_j)\left((1-\gamma)\sum_{i=0}^{n}\gamma^i d_\tau(s_k,s_j)\right) && (\text{induction hypothesis})
\end{aligned}$$

Now we use the lifting property: the only non-zero terms in the summation over $(s_k,s_j)$ are those for which $(s_k,s_j)\in\operatorname{supp}(\lambda_a)\subseteq\overset{\rho}{\sim}_{d_\tau(s,s')}$. Thus $(s_k,s_j)\in\overset{\rho}{\sim}_{d_\tau(s,s')}$, and we conclude that $d_\tau(s_k,s_j)=\inf_\varepsilon s_k\overset{\rho}{\sim}_\varepsilon s_j\leq d_\tau(s,s'),\forall s_k,s_j$.

$$\begin{aligned}
d_{n+1}(s,s') &\leq (1-\gamma)d_\tau(s,s') \\
&\qquad + \gamma\sum_{k,j}\lambda_a(s_k,s_j)\left((1-\gamma)\sum_{i=0}^{n}\gamma^i d_\tau(s,s')\right) \\
&= (1-\gamma)d_\tau(s,s')\sum_{i=0}^{n+1}\gamma^i
\end{aligned}$$

Which completes the induction. Taking limits finishes the proof:

$$d_\sim(s,s')\leq\frac{1-\gamma}{1-\gamma}d_\tau(s,s')=d_\tau(s,s'),$$

as desired. □

The bound is tight, and equality needs not hold, as Figure 3.2 shows. Consequently, using the bound from (Ferns, Panangaden, and Precup 2004, Theorem 5.1), the TE metric gives a guarantee on the difference in optimal value functions and on the approximation error for state-abstraction.

**Corollary 3.4.1.** *Let $\hat{V}$ be the value function in the abstract MDP of any abstraction $\phi$, not necessarily a bisimulation. Then, $\forall s, s' \in \mathcal{S}$:*

$$\begin{aligned} |V^*(s) - V^*(s')| &\leq \frac{1}{1-\gamma} d_\tau(\rho)(s, s') \quad \textit{and} \\ |\hat{V}^*(\phi(s)) - V^*(s)| &\leq \\ & \frac{\gamma}{(1-\gamma)^2} \max_{\phi(s')} \max_{s' \in \phi(s')} \frac{1}{|\phi(s')|} \sum_{s'' \in \phi(s)} d_\tau(\rho)(s', s'') \end{aligned}$$

### 3.4.2 Optimal Couplings

In Figure 3.2, the same coupling minimized both the bisimulation metric and the TE metric. Interestingly, the couplings chosen need not be the same in general. This is the content of the next theorem.

**Theorem 3.5.** *A minimum coupling $\lambda \in \operatorname{argmin}_{\Lambda(\mathcal{P}(\cdot|s,a), \mathcal{P}(\cdot|s',a))} \mathcal{W}(d_\sim)(\mathcal{P}(\cdot|s,a), \mathcal{P}(\cdot|s',a))$ of the bisimulation metric need not be a lifting of the optimal bisimulation $\overset{\rho}{\sim}_{d_\tau(\rho)(s,s')}$. Conversely, a coupling which lifts the optimal bisimulation $\overset{\rho}{\sim}_{d_\tau(\rho)(s,s')}$ need not be a minimizer of $\mathcal{W}(d_\sim)(\mathcal{P}(\cdot|s,a), \mathcal{P}(\cdot|s',a))$.*

*Proof.* Consider the following MDP with one action, taking $\rho(s, s') = \max_a |\mathcal{R}(s, a) - \mathcal{R}(s', a)|$ as our base metric.

And consider the following two couplings $\omega_\sim, \lambda_\tau \in \Lambda(p(s_0), p(t_0))$.

| $\omega_\sim$ | $s_1$ | $s_2$ |
|---|---|---|
| $t_1$ | $\varepsilon$ | $0$ |
| $t_2$ | $0$ | $1-\varepsilon$ |

| $\lambda_\tau$ | $s_1$ | $s_2$ |
|---|---|---|
| $t_1$ | $0$ | $\varepsilon$ |
| $t_2$ | $\varepsilon$ | $1-2\varepsilon$ |

We verify that $\omega_\sim$ minimizes the bisimulation distance and that $\lambda_\tau$ minimizes the tempo-

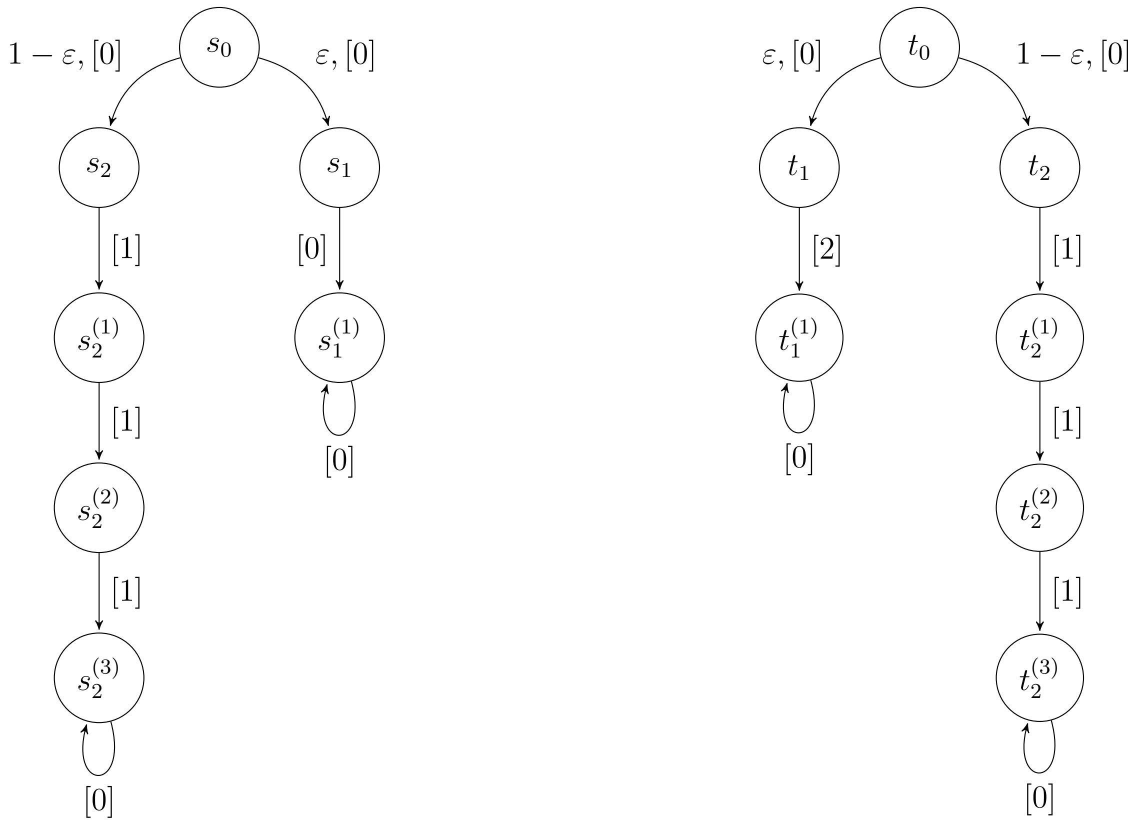

Figure 3.3: State $s_0$ transitions to $s_1, s_2$ with probability $\varepsilon, 1-\varepsilon$. Similarly for $t_0$. All other transitions are deterministic. Rewards indicated by square brackets.

rally extended metric. For $\omega_\sim$:

$$\begin{aligned}\langle \omega_\sim, d_\sim \rangle = \sum_{u,v\in\mathcal{S}} \omega_\sim(u,v) d_\sim(u,v) &= \varepsilon d_\sim(s_1,t_1) + (1-\varepsilon) d_\sim(s_2,t_2) \\ &= 2\varepsilon\{2(1-\gamma)\}.\end{aligned}$$

which is easily seen to be the minimizer of $\mathcal{W}(d_\sim)(\mathcal{P}(\cdot|s_0), \mathcal{P}(\cdot|t_0))$. Meanwhile, for $\lambda_\tau$ we have:

$$\begin{aligned}\langle \lambda_{\mathcal{T}}, d_\sim \rangle = \sum_{u,v \in \mathcal{S}} \lambda_{\mathcal{T}}(u,v) d_\sim(u,v) &= \varepsilon d_\sim(s_1,t_2) + \varepsilon d_\sim(s_2,t_1) \\ &\quad + (1-2\varepsilon) d_\sim(s_2,t_2) \\ &= \varepsilon d_\sim(s_1,t_2) + \varepsilon d_\sim(s_2,t_1) \\ &= 2\varepsilon\{(1-\gamma)(1+\gamma+\gamma^2)\} > \langle \omega_\sim, d_\sim \rangle\end{aligned}$$

On the other hand, using $\omega_\sim$, the best relation that can be lifted is $\overset{\rho}{\sim}_2$, since $(s_1,t_1) \in \operatorname{supp}(\omega_\sim)$ and $\mathcal{R}(s_1) - \mathcal{R}(t_1) = 2$. Meanwhile, $\lambda_{\mathcal{T}}$ achieves the minimum lifting of $\overset{\rho}{\sim}_1$, since $(s_1,t_2), (s_2,t_1), (s_2,t_2)$ all have reward differences of $1$. Thus $\omega_\sim$ minimizes the bisimulation metric but not the temporal extension metric. Conversely, $\lambda_{\mathcal{T}}$ minimizes the temporal extension metric since it achieves the minimum lifting, but not the bisimulation metric. □

**Remark 3.1** (Metric on liftings vs. lifting of metric)**.** This example the different behaviours of the two metrics – the TE metric aims to minimize the reward difference between coupled states at every step so as to ensure that a single bisimulation relation holds, whereas the bisimulation metric is not preserving a single relation and is willing to couple large differences at an initial step. The couplings chosen by the bisimulation metric do not give a (generalized) bisimulation relation, and the best that one can do with a (generalized) bisimulation relation is given by the temporal extension. This contrasts the two different lifting procedures. The temporal extension, which minimizes over $\varepsilon$-bisimulation relations, is a metric on a family of liftings of relations. On the other hand, the bisimulation metric is defined with respect to the Wasserstein metric, which is a lifting of metrics. In short: a metric on liftings does not equal a lifting of metrics.

## 3.5 Related Work

Generalizations of bisimulation relations to approximate bisimulation relations have appeared before in different forms and different contexts.

One notion of approximation bisimulations originates from the work of Desharnais et al., which introduces $\varepsilon$-simulation and $\varepsilon$-bisimulation relations for labelled Markov processes as opposed to Markov decision processes (Desharnais, Laviolette, and Tracol 2008). In addition to the different setting, the two definitions are quite different: their bisimulation relations are based on an approximate modal logic, and the "approximate" part of the approximate relations is with respect to the transition probabilities rather than a base metric. Furthermore, there is no notion of a base relation being lifted.

One definition which incorporates metrics for the use of approximate bisimulation is the work of Girard and Pappas, which defines $\varepsilon$-bisimulations in the context of deterministic and nondeterministic automata (Girard and Pappas 2007). Their definition for state-similarity metrics is similar to ours: namely, minimizing over a family of $\epsilon$-bisimulation relations. However, the definition of bisimulation for these models is not probabilistic and is based on matching transitions. Furthermore, in their setting the "base metric" is in the definition of the automata, rather than a flexible choice which is independent of the given problem.

In RL, the notion of *approximate* abstractions has been investigated in various ways (see (Abel, Hershkowitz, and Littman 2016) and references therein), and occasionally through the lens of bisimulation or homomorphism relations. However, as far as we can tell, the coupling view, the generalized notion of $\Phi$-bisimulation, and the definition of a pseudometric through minimization over quantitative bisimulations are novel contributions to the literature.

Finally, we note that the connection between liftings and the definition based on equivalence classes has been observed before in the different setting considered by the concurrency theory community (Deng and Du 2011). Our results extend this characterization to RL as the definition of bisimulation for MDPs includes the equality of rewards condition for the first step. The notion of bisimulation considered in that literature only focuses on the transition distribution aspect (condition 2 of the definition of bisimulation for MDPs). More generally, our lifting-based definition of $\Phi$-bisimulation differs from previous definitions due to requiring that the states are related via a relation $\Phi$ at every step in addition to satisfying the lifting criterion.

## 3.6 Discussion & Future Work

We have introduced the *temporally extended metrics*, a novel class of metrics for behavioural equivalence, which are based on a generalized notion of bisimulation. We have established bounds and other connections with the bisimulation metric, and seen that they neither compute the same values nor pick out the same couplings of state distributions. This work marks the beginning of an investigation into formally safe, computationally tractable, and model-free metrics for behavioural equivalence. There are many interesting avenues for future work that we intend to pursue. For computational aspects, the TE metric involves the computation of a bisimulation relation rather than a bisimulation metric, which can be done exactly in $\mathcal{O}(|\mathcal{A}||\mathcal{S}|^3)$ via partition refinements as opposed to approximated up to a degree of accuracy $\varepsilon$ in $\mathcal{O}(|\mathcal{A}||\mathcal{S}|^4 \log |\mathcal{S}| \log \varepsilon)$ (Ferns, Panangaden, and Precup 2004). Deriving an exact algorithm, however, is left for future work. The possibility of model-free computation is hypothesized since the metric requires only the *existence* of a lifting, as opposed finding the exact weights of an optimal coupling as does the Wasserstein metric, thus should be easier to estimate from samples.

Although the metrics are continuous with respect to reward parameters of the MDP, a slightly disconcerting aspect of the TE metrics is their discontinuity with respect to the transition distributions, as observed in Figure 3.2. This is because we require exact couplings - we are currently investigating the use of *approximate couplings* to remedy this, which have recently surfaced in the study of differential privacy (Barthe et al. 2016). In short, the marginals of approximate couplings are not required to match exactly, and can have a certain slack. This could allow us to capture discontinuities in the transitions distributions in addition to being able to capture them in the rewards.

Finally, as the general notions of $\Phi$-bisimulations and temporal extensions can be considered for arbitrary relations and metrics, examining the interplay between different notions of bisimulation (e.g. for optimal value functions or policy value functions) could be a fruitful direction.

# Chapter 4

# Bisimulation of algorithms: Distributional RL vs. Expected RL

In this short chapter, we will adapt the tools and insights from the temporally extended metrics to a new setting: rather than comparing the behaviours of two states of an MDP, we will compare the performance of initializations in different RL algorithms. The tools are readily applicable to this new setting: the algorithms are modelled as Markov processes on the space of value functions. The trajectory of an algorithm in this space is governed by the randomly sampled transitions and the update rule. We couple the sampling distributions of the two algorithms, and find that we can exhibit recursive couplings which support bisimulation relations.

The two classes of algorithms under consideration are Distributional RL and the usual expected-value RL. This chapter is inspired from the recent paper (Lyle, Castro, and Bellemare 2019). The same results concerning the equivalence of the two algorithms is derived in that paper, although with different methods. Our aim is to show the novel use of bisimulation and coupling techniques for the analysis of RL algorithms.

The rest of this chapter is organized as follows: Section 4.1 provides some background on the distributional RL. Section 4.2 introduces the Markov process which models the RL algorithms. Section 4.3 provides the equivalence results. Lastly, Section 4.4 discusses possible directions for future work.

## 4.1 Distributional RL

Rather than learning the expected discounted sum of returns, the distributional approach to RL (Bellemare, Dabney, and Munos 2017) instead attempts to model the full distribution of returns. We follow the notation of (Rowland et al. 2018): the distribution of returns is written $\eta^\pi(s,a) \in \mathscr{P}(\mathbb{R})$. If $Z^\pi$ is a random variable distributed according to $\eta^\pi$ then we have $\mathbb{E}[Z^\pi(s,a)] = Q^\pi(s,a)$ for each $(s,a) \in \mathcal{S} \times \mathcal{A}$. The analogous distributional Bellman equations are given by

$$\mathcal{T}^\pi_{\text{dist}} Z^\pi(s,a) = \mathcal{R}(s,a) + \gamma Z^\pi(S', A'), \tag{4.1}$$

where $S' \sim \mathcal{P}(\cdot|s,a), A' \sim \pi(\cdot|S')$. In terms of measures, we can equivalently write

$$\mathcal{T}^\pi_{\text{dist}} \eta^\pi(s,a) = \sum_{(s',a') \in \mathcal{X} \times \mathcal{A}} (f_{\mathcal{R}(s,a),\gamma})_\# \eta^{(s',a')} \pi(a'|s') \mathcal{P}(s'|s,a),$$

where $(f_{a,b})(x) := a + bx$ and $(f)_\# \eta(A) = \eta(f^{-1}(A))$ is the pushforward measure of $\eta$ under $f$.

The stochastic approximation algorithm that we will consider is the distributional analogue of SARSA, which is a mixture update of the measure $\eta(s,a)$ and a pushforward of $\eta(s_{k+1}, a_{k+1})$ for a sampled transition $(s_k, a_k, \mathcal{R}(s_k,a_k), s_{k+1}, a_{k+1})$:

$$\eta_{k+1}(s_k,a_k) = (1-\alpha_k)\eta_k(s_k,a_k) + \alpha_k (f_{\mathcal{R}(s_k,a_k),\gamma})_\# \eta_k(s_{k+1}, a_{k+1})$$

For simplicity we will only examine the case of tabular updates. Our results will hold for both synchronous and asynchronous sampling rules.

## 4.2 Value-based RL algorithms define Markov Processes on the Space of Value Functions

For what follows, we cast value-based RL algorithms as *Markov processes* on the space of value functions. To avoid distinguishing between methods for learning value functions and methods for learning action-value functions, we will let $\mathcal{X}$ be an arbitrary finite domain (for example $\mathcal{X} = \mathcal{S}$ for value functions or $\mathcal{X} = \mathcal{S} \times \mathcal{A}$ for action-value functions). We define $\mathcal{F} \subseteq [\mathcal{X} \to \mathbb{R}]$ as the space of functions under consideration, and $\Sigma$ a suitable $\sigma$-algebra on $\mathcal{F}$ (e.g. $\mathcal{B}(\mathbb{R}^{|\mathcal{S}|})$ or $\mathcal{B}(\mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}|})$). With many RL algorithms, the stochasticity of the algorithm depends only on the sampled transition and the random current estimate, thus these algorithms define Markov processes over the space of functions in $\mathcal{F}$:

$$\mathbb{P}\{f_{n+1}|f_n, f_{n-1}, ..., f_1, f_0\} = \mathbb{P}\{f_{n+1}|f_n\}.$$

We will write $K$ for the Markov kernel of the algorithms, given by $K(f_n, \mathcal{A}) = \mathbb{P}\left\{f_{n+1} \in \mathcal{A}|f_n\right\}$ where $\mathcal{A} \in \Sigma$ is a measurable set.[1] The probability of transitioning from $f_{\text{old}}$ to $f_{\text{new}}$ under the kernel is precisely the probability of sampling a transition which, when plugged into the update rule, results in $f_{\text{new}}$. The precise form of the kernel will depend on algorithmic details such as the definition of the target and the step-sizes. We will usually not need to work with precise descriptions of the kernel and it will be enough to consider them abstractly. For a given probability measure $\mu \in \mathscr{P}(\mathcal{F})$, we write $\mu K(\mathcal{A}) = \int_{\mathcal{F}} \mu(\mathrm{d}\theta) K(\theta, \mathcal{A})$ for the distribution of functions after one transition. We write $K^n$ for the Markov kernel after $n$ steps, given inductively by $K^n(\theta, \mathcal{A}) = \int_{\mathcal{F}} K^{n-1}(\theta, \mathrm{d}\theta')K(\theta', \mathcal{A})$. A probability measure $\psi$ is invariant for $K$ if $\psi K = \psi$. Lastly we note that we use the term Markov process since the space of functions is uncountable, although the kernels

[1] In general the kernels may be time-dependent although we will only need to consider the time-homogeneous case.

themselves are often *discrete* (i.e. a finite sum of Dirac measures) for finite MDPs. With this formalism, we can construct adequate couplings between parallel Markov processes to show that the algorithms are bisimilar in the sense of Definition 3.2.

## 4.3 Bisimilarity of Distributional RL and Expected RL

Similarly to the approach taken by (Lyle, Castro, and Bellemare 2019), we define an "equality in expectation" relation $\mathcal{E} \subseteq \underbrace{\mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}|}}_{\text{value functions}} \times \underbrace{\mathscr{P}(\mathbb{R})^{|\mathcal{S}|\times|\mathcal{A}|}}_{\text{value distributions}}$ by $Q\mathcal{E}\eta$ if $\mathbb{E}_{Z\sim\eta}[Z] = Q$. We will show that a simple coupling can be constructed which lifts this relation for all iterations of the algorithm execution. Due to our lifting-based characterization of bisimulation, this precisely means that our algorithms are behaviourally identical.

We first recall the coupling view of bisimulation given by Theorem 3.1, and apply it to the Markov Processes induced by SARSA and distributional SARSA. Let $K$ and $K_{\text{dist}}$ be the Markov kernels of SARSA and distributional SARSA.

**Theorem 4.1.** *A relation $\mathcal{U} \subseteq \mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}|} \times \mathscr{P}(\mathbb{R})^{|\mathcal{S}|\times|\mathcal{A}|}$ is a $\mathcal{E}$-bisimulation relation $\iff$ $Q\mathcal{U}\eta$ implies:*

*(1) $Q\mathcal{E}\eta$*

*(2) $K(Q,\cdot)\ (\mathcal{U})^{\#}\ K_{dist}(\eta,\cdot)$*

Our main result is that initializations with the same expectation will bisimilar in this sense.

**Theorem 4.2.** *In the tabular setting, Distributional SARSA is $\mathcal{E}$-bisimilar to expected SARSA if they have the same step-size and update states according to the same sampling rule (synchronous or otherwise). That is, if the initializations are such that $Q_0\mathcal{E}\eta_0$ then we have $Q_0 \overset{\mathcal{E}}{\sim} \eta_0$.*

*Proof.* Let $\mathcal{U} = \{(Q, \eta) \mid Q\mathcal{E}\eta\}$ be the relation which relates action-value functions $Q$ to the distributions $\eta$ which have mean $Q$. Suppose $Q_0\mathcal{E}\eta_0$. To show the $\mathcal{E}$-bisimilarity, we have to exhibit a coupling of the transition kernels which supports the relation $\mathcal{U}$. For each $(s, a)$ which is updated we couple the algorithms to take the same samples:

$$\left.\begin{aligned} Q_1(s,a) &= (1-\alpha)Q_0(s,a) + \alpha\left(\mathcal{R}(s,a) + \gamma Q_0(s',a')\right) \\ \eta_1(s.a) &= (1-\alpha)\eta_0(s,a) + \alpha\left((f_{\mathcal{R}(s,a),\gamma})_{\#}\eta_0(s',a')\right) \end{aligned}\right\} \text{for the } \underline{\text{same}} \begin{aligned} s' &\sim p(\cdot|s,a), \\ a' &\sim \pi(\cdot|s') \end{aligned}$$

Evidently, the marginals match and this is a proper coupling. To check the support property we simply compute the expectation of $\eta_1$ and use the fact that the algorithms have sampled the same transitions. Writing $\eta^{(s,a)} := \eta(s,a)$:

$$\begin{aligned} \mathbb{E}_{Z_1(s,a)\sim\eta_1(s,a)}[Z_1(s,a)] &= \int_{\mathbb{R}} z\eta_1^{(s,a)}(\mathrm{d}z) \\ &= (1-\alpha)\int_{\mathbb{R}} z\eta_0^{(s,a)}(\mathrm{d}z) + \alpha\int_{\mathbb{R}} z(f_{\mathcal{R}(s,a),\gamma})_{\#}\eta_0(s',a')(\mathrm{d}z) \\ &= (1-\alpha)\mathbb{E}_{Z(s,a)\sim\eta_0(s,a)}[Z(s,a)] + \alpha\mathbb{E}_{Z_0(s',a')\sim\eta_0(s',a')}[\mathcal{R}(s,a) + \gamma Z_0(s',a')] \\ &= (1-\alpha)Q_0(s,a) + \alpha\left(\mathcal{R}(s,a) + \gamma Q_0(s',a')\right) = Q_1(s,a), \end{aligned}$$

where in the second equality we used the definition of the distributional SARSA update, in the third equality we have used that $\mathbb{E}_{Z\sim(f_{\mathcal{R},\gamma})_{\#}\eta}[Z] = \mathbb{E}_{Z\sim\eta}[\mathcal{R} + \gamma Z]$, and in the last equality we have used the definition of the expected SARSA update and the fact that $Q_1$ is also updated using $(s', a')$. So every pair of states $(Q_1, \eta_1)$ in the support of the coupling lies in $\mathcal{U}$, thus $\mathcal{U}$ is a bisimulation relation and we conclude that $Q_0 \overset{\mathcal{E}}{\sim} \eta_0$. □

## 4.4 Discussion & Future work

We have established the equivalence (in expectation) of distributional SARSA and expected SARSA for tabular methods. The method of proof was to define an appropriate relation ($\mathcal{E}$) and to construct a coupling which recursively supported that relation. The chosen coupling was very simple, we simply had the two algorithms sample the same state-action pairs. The result crucially used the coupling-based definition of bisimulation which was derived in Chapter 3.

As previously mentioned, these results were already derived in the recent (Lyle, Castro, and Bellemare 2019). However, as far as we are aware, the use of these proof methods is novel to the literature. As our focus was to illustrate the use of couplings and the proofs are similar, we have not focused on deriving further comparative results for the other updates rules which were considered in that paper.

In the same way that it is unlikely for two states to be perfectly bisimilar, we are unlikely to find other pairs of algorithms which are exactly equivalent. A natural extension would be to consider approximate bisimulation methods for more intricate analyses. An interesting direction of future work is to examine the use the bisimulation metrics or the temporally extended metrics of Chapter 3 to compare and analyze a broader class of algorithms.

# Chapter 5

# Distributional Stochastic Approximation: Convergence via Couplings

In the previous section, we saw how to use couplings (and a coupling-based characterization of bisimulation) to establish the equivalence of Distributional RL and Expected RL for certain settings. The coupling involved was very simple: simply couple the two algorithms to sample the same transitions. In this chapter, we will use the exact same coupling (which we call the same-sampling coupling) to establish the convergence of a wide class of value-based RL algorithms in one fell swoop. Rather than coupling two different algorithms with the same initialization and showing equivalence, we will instead couple *the same algorithm* with two different (arbitrary) initializations and establish that the updates are contractions on the space of distributions (with respect to the Wasserstein metric with the $\ell_\infty$-norm as a cost function).

Contraction arguments, in combination with the Banach Fixed Point Theorem, are the proverbial bread and butter of Dynamic Programming theory. Proofs of convergence for value iteration and policy iteration crucially hinge on the contractive properties of the Bellman operators $\mathcal{T}^\pi$ and $\mathcal{T}^\star$. In the sampling-based regime, however, proofs of convergence are much more intricate as one cannot simply use a contraction argument. This is not surprising – the updates taken by the algorithms depend on the particular sequence of samples observed and thus one has to incorporate probabilistic reasoning. For instance, typical stochastic approximation results for RL algorithms rely on hitting-time arguments to bound the sequence of random variables within progressively smaller regions (see, e.g., Bertsekas and Tsitsiklis 1996, Section 4.3).

In this section, we present a fresh perspective on stochastic approximation theory, which is to consider the evolution of the full distribution of possible function estimates. The setting for this analysis is the Markov process view introduced in Section 4.2. At the level of distributions, many commonly-used algorithms retain the contractive properties of their deterministic DP counterparts. In other words, while deterministic algorithms such as value iteration and policy evaluation are contractions on the space of functions, we show that sampling-based algorithms such as TD($\lambda$) and $Q$-learning are contractions on the space of distributions of functions. This enables quick and easy convergence proofs establishing weak convergence to a stationary distribution. We also characterize aspects of the stationary distribution obtained, and our characterization will hold for any algorithm whose target is in expectation a Bellman update.

Throughout this chapter, we will focus on the setting of synchronous updates and constant step-sizes. We provide avenues for extensions to the time-inhomogeneous case in Section 5.5.

## 5.1 A general convergence criterion

In this section, we provide a coupling-based criterion for determining if a given algorithm is a contraction on the space of distributions of functions. Recall from Section 4.2 that we write $\mathcal{F} \subseteq [\mathcal{X} \to \mathbb{R}]$ is the set of candidate functions on a finite domain $\mathcal{X}$. We will use the $\ell_\infty$ norm on $\mathcal{F}$ defined by $\|f\|_\infty = \max_x |f(x)|$ and will simply write $\|f\| \coloneqq \|f\|_\infty$. We will also use the shorthand $\mathcal{W}$ to denote the 1-Wasserstein metric with $\|\cdot\|$ as a cost function. Recall from Section 2.3 that most RL algorithms have an update rule of the form

$$\texttt{NewEstimate} \leftarrow (1 - \texttt{StepSize}) \times \texttt{OldEstimate} + \texttt{StepSize} \times \texttt{Target}.$$

We rewrite the target as a sampling term $\sigma(f, \omega)$, which takes as input a function and a random element $\omega \sim \mu$. The random element determines which sample is taken by the algorithm. The distribution $\mu$ captures the type of object which is being sampled (e.g., complete trajectories for Monte Carlo methods or single transitions for TD($0$)). Thus, we rewrite the update equation in the following way:

$$f_{n+1} \leftarrow (1 - \alpha) f_n + \alpha\sigma(f, \omega), \quad \omega \sim \mu. \tag{5.1}$$

In component-wise form, we have:

$$f_{n+1}(x) \leftarrow (1 - \alpha) f_n(x) + \alpha\sigma(f, \omega_x, x), \quad \omega_x \sim \mu_x \quad \forall\, x \in \mathcal{X}, \tag{5.2}$$

where the $\mu_x$ notation denotes that the sampling distribution for each $x$ may depend on $x$. It is easy to see that all the RL algorithms covered in Section 2.3 can be written in this manner. For example, SARSA updates for a policy $\pi$ can be written in this form with $\mathcal{X} = \mathcal{S} \times \mathcal{A}$, $\omega = (s', a')$ where $s' \sim \mathcal{P}(\cdot|s, a), a' \sim \pi(\cdot|s')$, and $\sigma(f, \omega, (s, a)) = \mathcal{R}(s, a) + \gamma f((s', a'))$. As another example, Monte Carlo Evaluation of value functions for a policy $\pi$ can be written with $\mathcal{X} = \mathcal{S}$, $\omega$ a random trajectory $(s_n, a_n)_{n \geq 0}$ collected starting at $s_0 = s$

and following $\pi$, and $\sigma(f, \omega, s) = \sum_t \gamma^t \mathcal{R}(s_t, a_t)$ the return of that trajectory. From the discussion in Section 4.2, each update rule induces a specific Markov kernel. We are now ready for the full statement of the theorem.

**Theorem 5.1** (Convergence Criterion)**.** *Let $K_\alpha$ be the Markov kernel induced by an update rule of the form* (5.1) *with step size $0 < \alpha < 1$, and $(f_k^\alpha)_{k \geq 0}$ a Markov chain generated by $K_\alpha$ and with arbitrary initialization $\lambda \in \mathcal{P}_1(\mathcal{F})$. Suppose that in the sampling term $\sigma(f, \omega)$, the sampling distribution $\mu$ does not depend on $f$. Further suppose that $\sigma(\cdot, \omega)$ is uniformly a contraction for each $\omega$, i.e. there exists a $\rho < 1$ such that for each $f^{(1)}, f^{(2)}$, and $\omega$:*

$$\left\|\sigma(f^{(1)}, \omega) - \sigma(f^{(2)}, \omega)\right\| \leq \rho \left\|f^{(1)} - f^{(2)}\right\|. \tag{5.3}$$

*Then the mapping $\lambda \mapsto \lambda K_\alpha$ is a contraction in $\mathcal{W}$ and in particular the sequence $(f_k^\alpha)_{k \geq 0}$ converges in distribution to a unique stationary distribution $\psi_\alpha \in \mathcal{P}_1(\mathcal{F})$.*

*Proof.* Let $\lambda^{(1)}, \lambda^{(2)} \in \mathcal{P}_1(\mathcal{F})$ be two initial distributions of function estimates. By Theorem 2.2, there exists a coupling $f_0^{(1)} \sim \lambda^{(1)}, f_0^{(2)} \sim \lambda^{(2)}$ which minimizes the transport cost, i.e. such that $\mathcal{W}(\lambda^{(1)}, \lambda^{(2)}) = \inf_{(X,Y)} \mathbb{E}\left[\|X - Y\|\right] = \mathbb{E}\left[\left\|f_0^{(1)} - f_0^{(2)}\right\|\right]$. We define the coupling $(f_1^{(1)}, f_1^{(2)})$ to take the same samples.

$$\left.\begin{aligned} f_1^{(1)} &= (1-\alpha) f_0^{(1)} + \alpha \sigma(f_0^{(1)}, \omega) \\ f_1^{(2)} &= (1-\alpha) f_0^{(2)} + \alpha \sigma(f_0^{(2)}, \omega) \end{aligned}\right\} \text{for the } \underline{\text{same}}\ \omega \sim \mu. \tag{5.4}$$

Note that $f_1^{(1)} \sim \lambda^{(1)} K_\alpha$ and $f_1^{(2)} \sim \lambda^{(2)} K_\alpha$ since the sampling distribution does not depend on the function estimates. Thus $(f_1^{(1)}, f_2^{(2)})$ is a valid coupling of $(\lambda^{(1)} K_\alpha, \lambda^{(2)} K_\alpha)$.

With this coupling, we will show that the map $\lambda \mapsto \lambda K_\alpha$ is a contraction mapping with respect to $\mathcal{W}$. Since $(f_1^{(1)}, f_1^{(2)})$ is a valid coupling, by definition of $\mathcal{W}$ we can upper bound

$\mathcal{W}(\lambda^{(1)}K_\alpha, \lambda^{(2)}K_\alpha) \leq \mathbb{E}\left[\left\|f_1^{(1)} - f_1^{(2)}\right\|\right]$. This gives:

$$\begin{aligned}
\mathcal{W}(\lambda^{(1)}K_\alpha, \lambda^{(2)}K_\alpha) &\leq \mathbb{E}\left[\left\|f_1^{(1)} - f_1^{(2)}\right\|\right] \\
&= \mathbb{E}\left[\left\|(1-\alpha)f_0^{(1)} + \alpha\sigma(f_0^{(1)}, \omega) - \left((1-\alpha)f_0^{(2)} + \alpha\sigma(f_0^{(2)}, \omega)\right)\right\|\right] \\
&\leq (1-\alpha)\mathbb{E}\left[\left\|f_0^{(1)} - f_0^{(2)}\right\|\right] + \alpha\mathbb{E}\left[\left\|\sigma(f_0^{(1)}, \omega) - \sigma(f_0^{(2)}, \omega)\right\|\right] \\
&\leq ((1-\alpha) + \alpha\rho)\,\mathbb{E}\left[\left\|f_0^{(1)} - f_0^{(2)}\right\|\right] && \text{(by (5.1))} \\
&= (1-\alpha+\alpha\rho)\,\mathcal{W}(\lambda^{(1)}, \lambda^{(2)})
\end{aligned}$$

Since $\rho < 1$ and $0 < \alpha < 1$ this implies $1 - \alpha + \alpha\rho < 1$, and thus that the kernel is a contraction. Since $\mathcal{P}_1$ metrized with $\mathcal{W}$ is a complete metric space (Theorem 2.3), from Banach's fixed point theorem it follows that $(\lambda K_\alpha^n)_{n\geq 0}$ converges to a unique fixed point $\psi_\alpha$ for any initial distribution $\lambda$. Finally $\psi_\alpha$ is a stationary distribution by the fixed point property:

$$\psi_\alpha \mathcal{D}_\alpha = \psi_\alpha \qquad \square$$

**Remark 5.1** (About the independence condition)**.** In the conditions of the theorem we have required that $\omega \sim \mu$ in the sampling term $\sigma(f, \omega)$ does not depend on $f$. This means that the sampled transitions can not depend on the current estimate of value function. This is not a restrictive assumption – in fact most RL algorithms sample transitions from the MDP based only on a fixed policy and the current state. An example of an algorithm which *does not* satisfy this assumption is the Optimistic Policy Iteration algorithm (Tsitsiklis 2002), in which trajectories are sampled according to a policy which is greedy with respect to the current estimate. In that scenario the same-sampling coupling which we used in Equation (5.1) is not allowed.

**Remark 5.2** (About the contraction condition)**.** In the conditions of the theorem we have also required that, for each $\omega$, $\sigma(\cdot, \omega)$ is a contraction. This is not a restrictive assumption.

We will provide some examples in the following sections. Some examples of targets which satisfy the assumptions are Monte Carlo returns or Temporal-Difference returns. In most examples, a sum of reward terms will cancel in the difference $\|\sigma(f_1, \omega) - \sigma(f_2, \omega)\|$ and we are left with a discount factor multiplied by $\|f_1 - f_2\|$. We will provide some examples in the following sections.

### 5.1.1 Proof strategy

In the statement of Theorem 5.1, we attempted to provide the most general conditions which would guarantee convergence. But of course it is not possible to capture every RL algorithm under one umbrella. Thus, we provide a simple proof recipe which can be used to show convergence under different scenarios:

**(P1)** Let $\lambda^{(1)}, \lambda^{(2)}$ be initial distributions and $(f_0^{(1)}, f_0^{(2)})$ be the optimal coupling which minimizes $\mathcal{W}(\lambda^{(1)}, \lambda^{(2)})$;

**(P2)** Define an appropriate coupling $f_1^{(1)} \sim \lambda^{(1)}K, f_1^{(2)} \sim \lambda^{(2)}K$ - e.g. by defining them to follow the same trajectories if the updates sample from the same distributions;

**(P3)** Use the upper bound $\mathcal{W}(\lambda^{(1)}K, \lambda^{(2)}K) \leq \mathbb{E}\left[\left\|f_1^{(1)} - f_2^{(2)}\right\|\right]$ and bound $\mathbb{E}\left[\left\|f_1^{(1)} - f_1^{(2)}\right\|\right] \leq \rho\mathbb{E}\left[\left\|f_0^{(1)} - f_0^{(2)}\right\|\right]$ for some $\rho < 1$ (usually follows from the recursive nature of the updates) to show that $\mu \mapsto \mu K$ is a contraction.

## 5.2 Examples: Monte Carlo, SARSA, Q-Learning, TD($\lambda$)

As applications of the general convergence criterion, we will show that convergence follows easily for a myriad of commonly-used algorithms.

### 5.2.1 Monte Carlo Evaluation

Recall from Chapter 2 that the Monte Carlo Evaluation updates are given by:

$$V_{k+1}(s) = (1-\alpha)V_k(s) + \alpha G_k^\pi(s), \tag{MCE}$$

where $G_k^\pi(s)$ is the return of a random trajectory collected at $s$ and following $\pi$. Here we have $\omega = (s_n, a_n)_{\geq 0}$ and $\sigma(f^{(1)}, \omega) = \sum_t \gamma^t \mathcal{R}(s_t, a_t)$. By theorem 5.1, we need only show that for the same-sampling coupling we have $\left\|\sigma(f_1^{(1)}, \omega) - \sigma(f_1^{(2)}, \omega)\right\|$. This follows trivially since $\sigma$ does not actually depend on $f$.

$$\left\|\sigma(f_1^{(1)}, \omega) - \sigma(f_1^{(2)}, \omega)\right\| = \max_s \left|\sum_t \gamma^t \mathcal{R}(s_t, a_t) - \sum_t \gamma^t \mathcal{R}(s_t, a_t)\right| = 0.$$

Since the two processes sample the same trajectories, the difference in the returns is 0.

### 5.2.2 TD(0)

The TD(0) updates are given by:

$$V_{k+1}^{(1)}(s) = (1-\alpha)V_k^{(1)}(s) + \alpha\left(\mathcal{R}(s,a) + \gamma V_k^{(1)}(s')\right) \qquad \begin{matrix} a \sim \pi(\cdot|s) \\ s' \sim \mathcal{P}(\cdot|s,a) \end{matrix} \tag{TD(0)}$$

Here we have that $\omega_s = (a_s, s'_s)$ (the subscript indicates dependence on $s$) and $\sigma(V, \omega_s, s) = \mathcal{R}(s, a_s) + \gamma V(s'_s)$. Again, the bound follows easily:

$$\begin{aligned}\left\|\sigma(V_1^{(1)}, \omega) - \sigma(V_1^{(2)}, \omega)\right\| &= \max_s |\mathcal{R}(s, a_s) + \gamma V^{(1)}(s'_s) - \mathcal{R}(s, a_s) - \gamma V^{(2)}(s'_s)| \\ &= \gamma \max_s |V^{(1)}(s'_s) - V^{(2)}(s'_s)| \\ &\leq \gamma \max_s |V^{(1)}(s) - V^{(2)}(s)| = \gamma \left\|V^{(1)} - V^{(2)}\right\|\end{aligned}$$

We have made crucial use of the fact that $V^{(1)}$ and $V^{(2)}$ have sampled the same transitions to obtain the inequality $\max_s |V^{(1)}(s'_s) - V^{(2)}(s'_s)| \leq \max_s |V^{(1)}(s) - V^{(2)}(s)|$.

**Remark 5.3.** We note that when $\alpha = 1$ the stationary distribution which is obtained is precisely the value distribution of Distributional RL! This follows from the distributional Bellman equation (4.1). For $\alpha \neq 1$, the stationary distribution over value functions will in general not be the same as the value distribution of Distributional RL.

### 5.2.3 SARSA

The SARSA updates are given by:

$$Q_{k+1}(s,a) = (1-\alpha)Q_k(s,a) + \alpha\left(\mathcal{R}(s,a) + \gamma Q_k(s',a')\right) \qquad \begin{array}{l} s' \sim \mathcal{P}(\cdot|s,a) \\ a' \sim \pi(\cdot|s') \end{array} \tag{SARSA}$$

Here we have $\omega_{(s,a)} = (s'_{(s,a)}, a'_{(s,a)})$ and $\sigma(Q, \omega_{(s,a)}, (s,a)) = \mathcal{R}(s,a) + \gamma Q(s',a')$. The bound follows along the same lines. We omit the subscripts on $s', a'$:

$$\begin{aligned} \left\|\sigma(Q^{(1)},\omega) - \sigma(Q^{(2)},\omega)\right\| &= \max_{s,a}\left|\mathcal{R}(s,a) - \mathcal{R}(s,a) + \gamma\left(Q^{(1)}(s',a') - Q^{(2)}(s',a')\right)\right| \\ &= \gamma \max_{s,a}\left|Q^{(1)}(s',a') - Q^{(2)}(s',a')\right| \\ &\leq \gamma \max_{s,a}\left|Q^{(1)}(s,a) - Q^{(2)}(s,a)\right| = \gamma\left\|Q^{(1)} - Q^{(2)}\right\| \end{aligned}$$

### 5.2.4 Q-Learning

The $Q$-Learning updates are:

$$Q_{k+1}(s,a) = (1-\alpha)Q_k(s,a) + \alpha\left(\mathcal{R}(s,a) + \gamma \max_{a'} Q_k(s',a')\right) \quad s' \sim \mathcal{P}(\cdot|s,a) \tag{Q-Learning}$$

The bound follows again, but with one additional step. We omit subscripts on $s'$.

$$\begin{aligned}\left\|\sigma(Q^{(1)},\omega)-\sigma(Q^{(2)},\omega)\right\| &= \max_{s,a}\left|\mathcal{R}(s,a)-\mathcal{R}(s,a)+\gamma\left(\max_{a'}Q^{(1)}(s',a')-\max_{a'}Q^{(2)}(s',a')\right)\right| \\ &= \gamma\max_{s,a}\left|\max_{a'}Q^{(1)}(s',a')-\max_{a'}Q^{(2)}(s',a')\right| \\ &\leq \gamma\max_{s,a}\max_{a'}\left|Q^{(1)}(s',a')-Q^{(2)}(s',a')\right| \\ &\leq \gamma\max_{s,a}\left|Q^{(1)}(s,a)-Q^{(2)}(s,a)\right| = \gamma\left\|Q^{(1)}-Q^{(2)}\right\|\end{aligned}$$

### 5.2.5 TD($\lambda$)

The TD($\lambda$) updates are:

$$V_{k+1}(s)=(1-\alpha)V_k(s)+\alpha(1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}\left(\sum_{i=0}^{n}\gamma^i\mathcal{R}(s_i,a_i)+\gamma^nV_k(s_n)\right), \qquad \text{(TD}(\lambda))$$

with $\omega_s=(a_0,s_1,a_2,...)$. By the coupling, the reward terms will cancel in every n-step trajectory. We write $G_n^{(i)}=\mathcal{R}(s,a_0)+...+\gamma^{n-1}\mathcal{R}(s_{n-1},a_{n-1})+\gamma^nV^{(i)}(s_n)$ for the $n$-step return:

$$
\begin{aligned}
\left\|\sigma(V^{(1)},\omega)-\sigma(V^{(2)},\omega)\right\| &= (1-\lambda)\max_s\left|\sum_{n=1}^{\infty}\lambda^{n-1}G_n^{(1)}-\sum_{n=1}^{\infty}\lambda^{n-1}G_n^{(2)}\right| \\
&= (1-\lambda)\max_s\left|\sum_{n=1}^{\infty}\lambda^{n-1}\left(G_n^{(1)}-G_n^{(2)}\right)\right| \\
&= (1-\lambda)\max_s\left|\sum_{n=1}^{\infty}\lambda^{n-1}\left(\gamma^n V^{(1)}(s_n)-\gamma^n V^{(2)}(s_n)\right)\right| \\
&\qquad\qquad\qquad\qquad\qquad\qquad\text{(reward terms cancel)} \\
&= (1-\lambda)\max_s\left|\sum_{n=1}^{\infty}\lambda^{n-1}\gamma^n\left(V^{(1)}(s_n)-V^{(2)}(s_n)\right)\right| \\
&\leq (1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}\gamma^n\max_s\left|\left(V^{(1)}(s_n)-V^{(2)}(s_n)\right)\right| \\
&\leq (1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}\gamma^n\max_s\left|V^{(1)}(s)-V^{(2)}(s)\right| \\
&= (1-\lambda)\gamma\frac{\lambda\gamma}{1-\lambda\gamma}\left\|V^{(1)}-V^{(2)}\right\|
\end{aligned}
$$

Of course, it is not possible to cover all existing RL algorithms, but the techniques shown in these examples are widely applicable.

## 5.3 Characterizing the stationary distributions

In this section we give some results which characterize the stationary distributions obtained from (5.1). We give closed forms for the mean and the covariance of the stationary distributions for any such algorithm. The only requirement will be that the expected target is a Bellman update of $\mathcal{T}^{\pi}$ or $\mathcal{T}^{\star}$, which is commonly a design decision in any RL algorithm. Thus this characterization is true both for evaluation and control algorithms.

### 5.3.1 Mean is the true value function

Firstly, we can show that the mean of the stationary distribution is indeed the true value function $V^\pi$ or $V^\star$. We will write everything in terms of the policy case, as the optimality case is recovered by taking $\pi = \pi^\star$.

**Theorem 5.2.** *Let $\psi_\alpha$ be the stationary distribution obtained by theorem 5.1 when running the recursion (5.1) with learning rate $\alpha$. Suppose that $\sigma(f, \omega)$ is, in expectation over $\omega$, a Bellman update of $f$, i.e. $\mathbb{E}_{\omega\sim\mu}[\sigma(f,\omega)] = \mathcal{T}^\pi f$ for any policy $\pi$. Then $\overline{f_\alpha} = f^\pi$, where $\overline{f_\alpha} = \mathbb{E}[f_\alpha]$ is the expected value of $f_\alpha \sim \psi_\alpha$ and $f^\pi$ is the fixed point of $\mathcal{T}^\pi$.*

*Proof.* Let $f_0$ be distributed according to $\psi_\alpha$. Rewriting equation (5.1):

$$f_1 = (1-\alpha)f_0 + \alpha\mathcal{T}^\pi f_0 + \alpha\xi(f_0), \tag{5.5}$$

where $\xi(f_0) = \sigma(f_0, \omega) - \mathcal{T}^\pi f_0$ is a zero-mean noise term. Taking expectations on both sides, and using that $f_1$ is also distributed according to $\psi_\alpha$ by stationarity and that $\mathbb{E}[\xi(f)] = 0$ for any $f$:

$$\begin{aligned}
\overline{f_\alpha} &= (1-\alpha)\overline{f_\alpha} + \alpha\mathbb{E}[\mathcal{T}^\pi f_0] \\
\alpha\overline{f_\alpha} &= \alpha\mathbb{E}[\mathcal{R}^\pi + \gamma\mathcal{P}^\pi f_0] \\
\overline{f_\alpha} &= \mathcal{R}^\pi + \gamma\mathcal{P}^\pi\mathbb{E}[f_0] \\
\overline{f_\alpha} &= \mathcal{T}^\pi\overline{f_\alpha}
\end{aligned}$$

And therefore $\overline{f_\alpha} = f^\pi$ since it is the unique fixed point of $\mathcal{T}^\pi$. □

### 5.3.2 Covariance

In addition, we can derive a closed-form for the covariance of the stationary distribution. Again, the results will hold whenever the expected target is a Bellman update of some policy. We first introduce some linear algebra notation. For vectors $x, y \in \mathbb{R}^n$, we write $x \otimes y \in \mathbb{R}^{n\times n}$ for the tensor product of $x$ and $y$, defined by $(x \otimes y)(i, j) = x(i)y(j)$ for $i, j \in [n]$. By extension, the tensor product for two matrices $M, N \in \mathbb{R}^{n\times n}$ is the mapping $\mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$ defined by $M \otimes N : P \mapsto MPN$. A useful identity which we will repeatedly use is that $(S \otimes T)(x \otimes y) = (Sx) \otimes (Ty)$. With this notation the covariance of a random vector $\vec{X}$ with mean $\mu$ can be written $\mathbb{E}\left[(\vec{X} - \mu)^{\otimes 2}\right]$.

**Theorem 5.3.** *Let $\psi_\alpha$ be the stationary distribution obtained by theorem 5.1 when running the recursion* (5.1) *with learning rate $\alpha$, and let $f_\alpha \sim \psi_\alpha$. Define $\mathcal{C}(f) := \mathbb{E}\left[\xi(f)^{\otimes 2}\right]$ to be the covariance of the noise term for a given function. Suppose that $\sigma(f, \omega)$ is, in expectation, a Bellman update of $f$, i.e. $\mathbb{E}_{\omega\sim\mu}\left[\sigma(f, \omega)\right] = \mathcal{T}^\pi f$ for any policy $\pi$. For any $\alpha > 0$, the covariance of $f_\alpha$ is given by*

$$\mathbb{E}\left[(f_\alpha - f^\pi)^{\otimes 2}\right] = \alpha^2 \left[I - ((1 - \alpha)I + \alpha\gamma P^\pi)^{\otimes 2}\right]^{-1} \int \mathcal{C}(v)\psi_\alpha(\mathrm{d}v)$$

We preface the proof with some useful identities.

**Lemma 5.1.** *In the same setup as Theorem 5.3:*

$$\mathbb{E}\left[(f_\alpha - f^\pi) \otimes (\mathcal{T}^\pi f_\alpha - f^\pi + \xi(f_0))\right] = (I \otimes \gamma P^\pi)\mathbb{E}\left[(f_\alpha - f^\pi)^{\otimes 2}\right]$$

*and*

$$\mathbb{E}\left[((\mathcal{T}^\pi f_\alpha - f^\pi) + \xi(f_\alpha))^{\otimes 2}\right] = \gamma^2 (P^\pi)^{\otimes 2}\mathbb{E}\left[(f_\alpha - f^\pi)^{\otimes 2}\right] + \int C(v)\psi_\alpha(\mathrm{d}v)$$

*Proof.* Let $f_0 \sim \psi_\alpha$, by (5.1) we have $f_1 = (1 - \alpha)f_0 + \alpha(\mathcal{T}^\pi f_0 + \xi(f_0))$ and $f_1 \sim \psi_\alpha$. Furthermore, the distribution of $f_0$ is independent of the distribution of $\omega \sim \mu$ by the conditions of Theorem 5.1. By independence,

$$\begin{aligned}\mathbb{E}\left[(f_0 - f^\pi) \otimes \xi(f_0)\right] &= \mathbb{E}_{f_0}\mathbb{E}_\omega\left[(f_0 - f^\pi) \otimes \xi(f_0)\right] && \text{(by independence of } f_0 \text{ and } \xi(\cdot)\text{)}\\ &= \mathbb{E}_{f_0}\left[(f_0 - f^\pi) \otimes \mathbb{E}_\omega\xi(f_0)\right] = 0 && (\mathbb{E}_\omega[\xi(f)] = 0 \text{ for every } f)\end{aligned}$$

For the first identity, note that

$$\begin{aligned}\mathbb{E}\left[(f_0 - f^\pi) \otimes (\mathcal{T}^\pi f_0 - f^\pi)\right] &= \mathbb{E}\left[(f_0 - f^\pi) \otimes (\mathcal{R}^\pi + \gamma\mathcal{P}^\pi(f_0) - \mathcal{R}^\pi - \gamma\mathcal{P}^\pi(f^\pi)\right] \\ &= \mathbb{E}\left[(f_0 - f^\pi) \otimes \gamma\mathcal{P}^\pi(f_0 - f^\pi)\right] \\ &= \mathbb{E}\left[(I \otimes \gamma\mathcal{P}^\pi)(f_0 - f^\pi)^{\otimes 2}\right] = \gamma(I \otimes \mathcal{P}^\pi)\mathbb{E}\left[(f_0 - f^\pi)^{\otimes 2}\right].\end{aligned}$$

The first identity then follows by using $\mathbb{E}\left[(f_0 - f^\pi) \otimes \xi(f_0)\right] = 0$ and linearity of tensors products and expectations.

For the second identity, expanding the tensor product gives:

$$\begin{aligned}\mathbb{E}\left[((\mathcal{T}^\pi f_0 - f^\pi) + \xi(f_0))^{\otimes 2}\right] &= \mathbb{E}\Big[(\mathcal{T}^\pi f_0 - f^\pi)^{\otimes 2} + (\xi(f_0))^{\otimes 2}\Big] \\ &\quad + \mathbb{E}\left[\cancel{(\mathcal{T}^\pi f_0 - f^\pi) \otimes \xi(f_0)} + \cancel{\xi(f_0) \otimes (\mathcal{T}^\pi f_0 - f^\pi)}\right] \\ &= \mathbb{E}\left[(\gamma P^\pi(f_0 - f^\pi))^{\otimes 2}\right] + \int \mathcal{C}(v)\psi_\alpha(\mathrm{d}v) \\ &= (\gamma P^\pi)^{\otimes 2}\mathbb{E}\left[(f_0 - f^\pi)^{\otimes 2}\right] + \int \mathcal{C}(v)\psi_\alpha(\mathrm{d}v)\end{aligned}$$

where we used $\mathbb{E}\left[(\mathcal{T}^\pi f_0 - f^\pi) \otimes \xi(f_0)\right] = 0$. □

*Proof (of Theorem 5.3).* Again let $f_0$ be distributed according to $\psi_\alpha$. Subtracting $f^\pi$ from equation (5.5),

$$f_1 - f^\pi = (1-\alpha)(f_0 - f^\pi) + \alpha\left(\mathcal{T}^\pi f_0 - f^\pi + \xi(f_0)\right).$$

and taking tensor products:

$$\begin{aligned}(f_1 - f^\pi)^{\otimes 2} =& (1-\alpha)^2 (f_0 - f^\pi)^{\otimes 2} + \alpha^2 \left[\mathcal{T}^\pi f_0 - f^\pi + \xi(f_0)\right]^{\otimes 2} \\ &+ \alpha(1-\alpha)\left[(f_0 - f^\pi) \otimes (\mathcal{T}^\pi f_0 - f^\pi + \xi(f_0))\right] \\ &+ \alpha(1-\alpha)\left[(\mathcal{T}^\pi f_0 - f^\pi + \xi(f_0)) \otimes (f_0 - f^\pi)\right].\end{aligned}$$

Taking expectations on both sides, and using Lemma 5.1:

$$\begin{aligned}\mathbb{E}\left[(f_1 - f^\pi)^{\otimes 2}\right] =&(1-\alpha)^2\mathbb{E}\left[(f_0 - f^\pi)^{\otimes 2}\right] + \alpha^2\left\{(\gamma P^\pi)^{\otimes 2}\mathbb{E}[(f_0 - f^\pi)] + \int \mathcal{C}(v)\psi_a(\mathrm{d}v)\right\}\\ &+ \alpha(1-\alpha)\left(I \otimes \gamma P^\pi + \gamma P^\pi \otimes I\right)\mathbb{E}\left[(f_0 - f^\pi)^{\otimes 2}\right]\end{aligned}$$

Since $\mathbb{E}\left[(f_1 - f^\pi)^{\otimes 2}\right] = \mathbb{E}\left[(f_0 - f^\pi)^{\otimes 2}\right]$ by stationarity, re-arranging to the LHS and factoring gives:

$$\left[I - ((1-\alpha)I + \alpha\gamma P^\pi)^{\otimes 2}\right]\mathbb{E}\left[(f_0 - f^\pi)^{\otimes 2}\right] = \alpha^2 \int \mathcal{C}(v)\psi_\alpha(\mathrm{d}v)$$

We further show that the matrix on the LHS is invertible. By (Puterman 2014, Corollary C.4) it will follow from showing that $\rho\left(((1-\alpha)I + \alpha\gamma P^\pi)^{\otimes 2}\right) < 1$, where $\rho(A)$ is the spectral radius of matrix $A$. Writing $\|A\|_{\text{op}} = \max_i \sum_j A(i,j)$ for the operator norm of a matrix $A$, and using that $\rho(A) \leq \|A\|_{\text{op}}$, $\|A \otimes B\|_{\text{op}} = \|A\|_{\text{op}}\|B\|_{\text{op}}$, and $\|P^\pi\|_{\text{op}} = \|I\|_{\text{op}} = 1$:

$$\left\|((1-\alpha)I + \alpha\gamma P^\pi)^{\otimes 2}\right\|_{\text{op}} = \|(1-\alpha)I + \alpha\gamma P^\pi\|_{\text{op}}^2 \leq ((1-\alpha) + \alpha\gamma)^2 < 1 \quad \text{(since } \gamma < 1\text{)}$$

□

**Remark 5.4.** We note that the previous two results can be extended to a more general setting: the expectation of the target can be any operator $\mathcal{H}$ which has a fixed point $f_{\mathcal{H}}$ and commutes with the expectations (i.e. $\mathbb{E}[\mathcal{H}f] = \mathcal{H}\mathbb{E}[f]$). Theorems 5.2 and 5.3 hold mutantis mutandis. An example of such an operator would be the one considered in the Retrace($\lambda$) algorithm (Munos et al. 2016).

## 5.4 Related Work

The algorithms considered here have previously been shown to converge, often in numerous ways (e.g. for $Q$-learning, see Watkins and Dayan 1992; Tsitsiklis 1994; Jaakkola, Jordan, and Singh 1994; Littman and Szepesvári 1996). However, these results all focus on almost sure convergence of the iterates when the step sizes are taken to decrease according to the Robbins-Monro conditions ($\sum_t \alpha_t \to +\infty$ and $\sum_t \alpha_t^2 < +\infty$).

Weak convergence properties of stochastic approximation algorithms have also been examined. A classic reference on stochastic approximation is (Kushner and Yin 2003), which develops some theory for the constant step-size case in Chapter 8. Other works which study constant step-size algorithms are (Yu 2016; Lakshminarayanan and Szepesvári 2017). However, our results and methods are vastly different. In particular, the above references do not discuss convergence to a stationary distribution, and their methods of proof are not as simple. As far as we are aware, the result that RL algorithms are contractions on the space of distributions is novel.

Our methods for this section are similar to (Dieuleveut, Durmus, and Bach 2017), which develops the theory for constant-step size stochastic gradient descent in the context of supervised learning. In particular the proof of theorem 5.1 is inspired from their proposition 2, although simplified and adapted to the reinforcement learning setting.

## 5.5 Discussion & Future Work

In this chapter, we proposed a distributional perspective on stochastic approximation theory, which involves studying the evolution of the distribution of possible value function estimates. While DP theory relies fundamentally on contraction arguments, in RL (where algorithms are stochastic) similar techniques have not been immediately available. We showed that stochastic approximation algorithms for RL can indeed be found to be contractive if we "lift" to the space of distributions of functions.

We outlined general criteria for convergence, which are satisfied for many algorithms. There exist algorithms where the conditions of theorem 5.1 do not hold (e.g. Optimistic

Policy Iteration), although we are confident that the coupling methodology can extend. Indeed, we have only made use of one very simple coupling (the same-sampling coupling), so in a sense we are just scratching the surface. Avenues for future work include:

**Online updates/Decreasing step sizes**

With online updates, only the state which was most recently sampled gets updated. In the online setting or with changing step-sizes, we no longer have a time-homogeneous Markov kernel $K$. Instead we are now dealing with time-dependent kernels $K_n(f, A) = \mathbb{P}\{f_{n+1} \in A | f_n = f\}$. In particular, we will need a 'time-dependent' analogue of the Banach fixed point strategy to show $\lambda K_1 K_2 \cdots K_n \xrightarrow{n\to\infty} \psi$. This can be achieved, e.g., by showing that the sequence $(\lambda K_1 \cdots K_n)_{n\geq 0}$ is contractive and therefore Cauchy.[1] Time-dependent maps have been studied in the stochastic approximation literature (Bertsekas and Tsitsiklis 1996, Proposition 4.5), so it is possible that our methods provide similar simplifications to this setting as well.

**Optimistic Policy Iteration**

The Optimistic Policy Iteration algorithm (with synchronous updates and constant step-size) performs the following updates:

$$V_{n+1}(s) = (1-\alpha)V_n(s) + \alpha G^{\pi_{V_n}}(s),$$

where $\pi_{V_n}$ is a policy which is greedy with respect to $V_n$ and $G^{\pi_{V_n}}$ is the return of a random trajectory starting at $s$ and following $\pi_{V_n}$. The policy (and thus the sampling distribution) updates at every iteration and depends on the current value function. For two initial distributions $V_0^{(1)} \sim \lambda^{(1)}, V_0^{(2)} \sim \lambda^{(2)}$, we cannot couple $V_1^{(1)}$ and $V_1^{(2)}$ to sample the same

[1] A sequence is contractive if it satisfies $\mathcal{W}_2(\lambda K_1 \cdots K_n, \lambda K_1 \cdots K_{n+1}) \leq \rho \mathcal{W}_2(\lambda K_1 \cdots K_{n-1}, \lambda K_1 \cdots K_n)$

trajectories, since the sampling distribution depends on the value functions themselves. In DP theory, another fundamental property of the Bellman operators is monotonicity. Continuing the analogy between DP theory on functions and RL theory on distributions of functions, we are currently investigating the use of "monotonicity" properties for this algorithm. This requires defining a suitable notion of monotonicity between distributions. A good candidate is *stochastic domination*, which has deep ties with coupling theory (e.g. via Strassen's theorem) (Lindvall 2002, Chapter IV).

### Function approximation

So far, we have only considered tabular methods. It would be interesting to see if the methods extend to function approximation settings.

### Beyond value-based methods

Finally, beyond value-based methods, virtually every class of RL method features some randomness. As we can couple any type of stochastic process, making use of coupling techniques to reason about these other families of RL methods (e.g. policy gradient methods) promises to be a fruitful direction.

# Chapter 6

# Conclusion

In this thesis we presented numerous applications demonstrating the effectiveness of couplings as a tool for behavioural analyses in RL.

Our first application was the characterization of bisimulation via couplings (Section 3.1), or more specifically probabilistic liftings, and the subsequent generalization of this characterization to consider arbitrary relations between states instead of reward equality (Section 3.2) . We examined the case of quantitative relations arising from a base pseudometric of choice, and defined a notion of temporally extended metrics, which extend the base metric to reflect long-term differences in the MDP (Section 3.3). We compared and contrasted with previously-defined metrics for state-similarity, the bisimulation metrics, by providing bounds and examining their different behaviours on some simple MDPs (Section 3.4).

Next, we turned from the problem of measuring state similarity to the problem of measuring algorithm similarity, and showed that these were two sides of the same coin. By casting common RL algorithms as Markov processes on the space of value functions (Section 4.2), we are able to use the same coupling methods and insights acquired thus far. By exhibiting a very simple coupling (the same-sampling coupling), we provide an example of algorithms which are *bisimilar* for the "equality in expectation" relation: namely, the distributional SARSA algorithm and the usual expected-value SARSA algorithm in the tabular setting (Section 4.3).

Lastly, we turned from measuring the long-term behaviours of two algorithms with similar initializations to measuring that of the same algorithm with differing initializations. For the synchronous and constant step-size case, we consider the evolution of the full distribution of possible value functions estimates of common sampling-based algorithms, and, with the aid of the same-sampling coupling, show that many common RL algorithms are contractions on the space of such distributions. We provide a simple criteria for verifying if a given update rule induces a contraction (Section 5.1), which is easily checked for a variety of cases (Section 5.2). The contraction property entails the convergence of these algorithms to a stationary distribution, which we characterize by providing closed-form expressions for the mean and covariance (Section 5.3).

For each of these developments, we have highlighted promising avenues for future work (see Section 3.6, Section 4.4, and Section 5.5, respectively).

# Bibliography

Abel, David, David Hershkowitz, and Michael Littman (2016). “Near Optimal Behavior via Approximate State Abstraction”. In: *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, pp. 2915–2923.

Arjovsky, Martin, Soumith Chintala, and Léon Bottou (2017). “Wasserstein gan”. In: *arXiv preprint arXiv:1701.07875*.

Banach, Stephen (1922). “Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales”. In: *Fund. Math. 3(1922)*.

Barthe, Gilles et al. (2016). “Proving differential privacy via probabilistic couplings”. In: *2016 31st Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*. IEEE, pp. 1–10.

Bellemare, Marc G, Will Dabney, and Rémi Munos (2017). “A distributional perspective on reinforcement learning”. In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, pp. 449–458.

Bellman, Richard (1966). “Dynamic programming”. In: *Science* 153.3731, pp. 34–37.

Bertsekas, Dimitri P and John N Tsitsiklis (1996). *Neuro-dynamic programming*. Vol. 5. Athena Scientific Belmont, MA.

Deng, Yuxin and Wenjie Du (2011). “Logical, metric, and algorithmic characterisations of probabilistic bisimulation”. In: *arXiv preprint arXiv:1103.4577*.

Desharnais, J. et al. (1999). “Metrics for Labeled Markov Systems”. In: *Proceedings of CONCUR99*. Lecture Notes in Computer Science 1664. Springer-Verlag.

Desharnais, Josée, François Laviolette, and Mathieu Tracol (2008). “Approximate analysis of probabilistic processes: Logic, simulation and games”. In: *2008 Fifth International Conference on Quantitative Evaluation of Systems*. IEEE, pp. 264–273.

Dieuleveut, Aymeric, Alain Durmus, and Francis Bach (2017). “Bridging the gap between constant step size stochastic gradient descent and markov chains”. In: *arXiv preprint arXiv:1707.06386*.

Ding, Ding et al. (2018). “Toward detecting violations of differential privacy”. In: *arXiv preprint arXiv:1805.10277*.

Ferns, Norm, Prakash Panangaden, and Doina Precup (2004). “Metrics for Finite Markov Decision Precesses”. In: *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, pp. 162–169.

Ferns, Norm and Doina Precup (2014). “Bisimulation Metrics are Optimal Value Functions”. In: *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence*. Article 67.

Ghavamzadeh, Mohammad et al. (2011). “Speedy Q-Learning”. In: *Advances in Neural Information Processing Systems 24*. Ed. by J. Shawe-Taylor et al. Curran Associates, Inc., pp. 2411–2419.

Girard, Antoine and George J Pappas (2007). “Approximation metrics for discrete and continuous systems”. In: *IEEE Transactions on Automatic Control* 52.5, pp. 782–798.

Givan, Robert, Thomas Dean, and Matthew Greig (2003). “Equivalence notions and model minimization in Markov decision processes”. In: *Artificial Intelligence* 147.1-2, pp. 163–223.

Hsu, Justin (2017). "Probabilistic Couplings for Probabilistic Reasoning". PhD thesis. University of Pennsylvania.

Jaakkola, Tommi, Michael I Jordan, and Satinder P Singh (1994). "Convergence of stochastic iterative dynamic programming algorithms". In: *Advances in neural information processing systems*, pp. 703–710.

Kushner, Harold and G George Yin (2003). *Stochastic approximation and recursive algorithms and applications*. Vol. 35. Springer Science & Business Media.

Lakshminarayanan, Chandrashekar and Csaba Szepesvári (2017). "Linear stochastic approximation: Constant step-size and iterate averaging". In: *arXiv preprint arXiv:1709.04073*.

Larsen, K. G. and A. Skou (1991). "Bisimulation through Probablistic Testing". In: *Information and Computation* 94, pp. 1–28.

Li, Lihong, Thomas J Walsh, and Michael L Littman (2006). "Towards a unified theory of state abstraction for MDPs." In:

Lindvall, Torgny (2002). *Lectures on the coupling method*. Courier Corporation.

— (1999). "On Strassen's theorem on stochastic domination". In: *Electronic communications in probability* 4, pp. 51–59.

Littman, Michael L and Csaba Szepesvári (1996). "A generalized reinforcement-learning model: Convergence and applications". In:

Lyle, Clare, Pablo Samuel Castro, and Marc G. Bellemare (2019). "A Comparative Analysis of Expected and Distributional Reinforcement Learning". In: *CoRR* abs/1901.11084.

Milner, R. (1980). *A Calculus for Communicating Systems*. Vol. 92. Lecture Notes in Computer Science. Springer-Verlag.

Munos, Rémi et al. (2016). "Safe and efficient off-policy reinforcement learning". In: *Advances in Neural Information Processing Systems*, pp. 1054–1062.

Park, D. (1981). “Concurrency and automata on infinite sequences”. In: *Proceedings of the 5th GI Conference on Theoretical Computer Science*. Lecture Notes In Computer Science 104. Springer-Verlag, pp. 167–183.

Puterman, Martin L (2014). *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

Rowland, Mark et al. (2018). “An analysis of categorical distributional reinforcement learning”. In: *arXiv preprint arXiv:1802.08163*.

Sutton, Richard S and Andrew G Barto (1998). *Introduction to reinforcement learning*. Vol. 135. MIT press Cambridge.

Thorisson, Hermann (2000). *Coupling, Stationarity and Regeneration*.

Tsitsiklis, John N (1994). “Asynchronous stochastic approximation and Q-learning”. In: *Machine learning* 16.3, pp. 185–202.

— (2002). “On the convergence of optimistic policy iteration”. In: *Journal of Machine Learning Research* 3.Jul, pp. 59–72.

Villani, Cédric (2008). *Optimal transport: old and new*. Springer-Verlag.

Watkins, Christopher JCH and Peter Dayan (1992). “Q-learning”. In: *Machine learning* 8.3-4, pp. 279–292.

Yu, Huizhen (2016). “Weak convergence properties of constrained emphatic temporal-difference learning with constant and slowly diminishing stepsize”. In: *The Journal of Machine Learning Research* 17.1, pp. 7745–7802.