Representations for Stable Off-Policy Reinforcement Learning

Dibya Ghosh [1] **Marc G. Bellemare** [1]

Abstract

Reinforcement learning with function approximation can be unstable and even divergent, especially when combined with off-policy learning and Bellman updates. In deep reinforcement learning, these issues have been dealt with empirically by adapting and regularizing the representation, in particular with auxiliary tasks. This suggests that representation learning may provide a means to guarantee stability. In this paper, we formally show that there are indeed nontrivial state representations under which the canonical TD algorithm is stable, even when learning off-policy. We analyze representation learning schemes that are based on the transition matrix of a policy, such as proto-value functions, along three axes: approximation error, stability, and ease of estimation. In the most general case, we show that a Schur basis provides convergence guarantees, but is difficult to estimate from samples. For a fixed reward function, we find that an orthogonal basis of the corresponding Krylov subspace is an even better choice. We conclude by empirically demonstrating that these stable representations can be learned using stochastic gradient descent, opening the door to improved techniques for representation learning with deep networks.

1. Introduction

Value function learning algorithms are known to demonstrate divergent behavior under the combination of bootstrapping, function approximation, and off-policy data, what Sutton & Barto (2018) call the "deadly triad" (see also van Hasselt et al., 2018). In reinforcement learning theory, it is well-established that methods such as Q-learning and TD(0) enjoy no general convergence guarantees under linear function approximation and off-policy data (Baird, 1995; Tsitsiklis & Roy, 1996). Despite this potential for failure, Q-learning and other temporal-difference algorithms remain the methods of choice for learning value functions in practice due to their simplicity and scalability.

In deep reinforcement learning, instability has been mitigated empirically through the use of auxiliary tasks, which shape and regularize the representation that is learned by the neural network. Methods using auxiliary tasks concurrently optimize the value function loss and an auxiliary representation learning objective such as visual reconstruction of observation (Jaderberg et al., 2016), latent transition and reward prediction (Gelada et al., 2019), adversarial value functions (Bellemare et al., 2019), or inverse kinematics (Pathak et al., 2017). In robotics, distributional reinforcement learning (Bellemare et al., 2017) in particular has proven a surprisingly effective auxiliary task (Bodnar et al., 2019; Vecerik et al., 2019; Cabi et al., 2019). While the stability of such methods remains an empirical phenomenon, it suggests that a carefully chosen representation learning algorithm may provide a means towards formally guaranteed stability of value function learning.

In this paper, we seek procedures for discovering representations that guarantee the stability of TD(0), a canonical algorithm for estimating the value function of a policy. We analyze the expected dynamics of TD(0), with the aim of characterizing representations under which TD(0) is provably stable. Learning dynamics of temporal-difference methods have been studied in depth in the context of a fixed state representation (Tsitsiklis & Roy, 1996; Borkar & Meyn, 2000; Yu & Bertsekas, 2009; Maei et al., 2009; Dalal et al., 2017). We go one step further by considering this representation as a component that can actively be shaped, and study stability guarantees that emerge from various representation learning schemes.

We show that the stability of a state representation is affected by: 1) the space of value functions it can express, and 2) how it parameterizes this space. We find a tight connection between stability and the geometry of the transition matrix, enabling us to provide stability conditions for algorithms that learn features from the transition matrix of a policy (Dayan, 1993; Mahadevan & Maggioni, 2007; Wu et al., 2018; Behzadian et al., 2019) and rewards (Petrik, 2007; Parr et al., 2007). Our analysis reveals that a number of

[1]Google Research. Correspondence to: Dibya Ghosh <dibya.ghosh@berkeley.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

popular representation learning algorithms, including proto-value functions, generally lead to representations that are not stable, despite their appealing approximation characteristics.

As special cases of a more general framework, we study two classes of stable representations. The first class consists of representations that are approximately invariant under the transition dynamics (Parr et al., 2008), while the second consists of representations that remain stable under reparameterization. From this study, we find that stable representations can be obtained from common matrix decompositions and furthermore, as solutions of simple iterative optimization procedures. Empirically, we find that different procedures trade off learnability, stability, and approximation error. In the large data regime, the Schur decomposition and a variant of the Krylov basis (Petrik, 2007) emerge as reliable techniques for obtaining a stable representation.

We conclude by demonstrating that these techniques can be operationalized using stochastic gradient descent on losses. We show that the Schur decomposition arises from the task of predicting the expectation of one's own features at the next time step, whereas a variant of the Krylov basis arises as from the task of predicting future expected rewards. This is particularly significant, as both of these auxiliary tasks have in fact been heuristically proposed in prior work (François-Lavet et al., 2018; Gelada et al., 2019). Our result confirms the validity of these auxiliary tasks, not only for improving approximation error but, more importantly, for taming the famed instabilities of off-policy learning.

2. Background

We consider a Markov decision process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r, \rho, \gamma)$ on a finite state space $\mathcal{S}$ and finite action space $\mathcal{A}$. The state transition distribution is given by $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$, the reward function $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, the initial state distribution $\rho \in \Delta(\mathcal{S})$, and the discount factor $\gamma \in [0, 1)$. We write $\mathcal{H} = \mathcal{S} \times \mathcal{A}$ with $|\mathcal{H}| = n$, and treat real-valued functions of state and action as vectors in $\mathbb{R}^n$.

A stochastic policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ induces a Markov chain on $\mathcal{H}$ with transition matrix $P^\pi \in \mathbb{R}^{n \times n}$. The value function $Q^\pi \in \mathbb{R}^n$ for a policy π is the expected return conditioned on the starting state-action pair,

$$Q^\pi(s_i, a_i) = \mathbb{E}_\pi \left[\sum_{t \geq 0} \gamma^t r(s_t, a_t) \,|\, s_0 = s_i, a_0 = a_i \right].$$

The value function also satisfies Bellman's equation; in vector notation (Puterman, 1994),

$$Q^\pi = r + \gamma P^\pi Q^\pi$$

from which we recover the concise $Q^\pi = (I - \gamma P^\pi)^{-1} r$.

2.1. Approximate Policy Evaluation

Approximate policy evaluation is the problem of estimating Q^π from a family of value functions $\{Q_\theta\}_{\theta \in \mathbb{R}^d}$ given a distribution of transitions $(s, a, r, s') \sim \xi(s, a) P(s'|s, a)$ (c.f. Bertsekas, 2011). We refer to $\xi \in \Delta(\mathcal{H})$ as the *data distribution*, and define $\Xi \in \mathbb{R}^{n \times n}$ a diagonal matrix with the elements of ξ on the diagonal. If the data distribution is the stationary distribution of P^π, the data is *on-policy* and *off-policy* otherwise. We equip $\mathbb{R}^n$ with the inner product and norm that is induced by the data distribution: $\langle v_1, v_2 \rangle_\Xi = v_1^\top \Xi v_2$. Most concepts from Euclidean inner products extend to this setting; see Appendix A for a review.

We consider a two-stage procedure for estimating value functions (Levine et al., 2017; Chung et al., 2019; Bertsekas, 2018). We first learn a *representation*, a d-dimensional mapping $\phi : \mathcal{H} \to \mathbb{R}^d$, through an explicit representation learning step. After a representation is learned, approximate policy evaluation is performed with the family of value functions linear in the representation ϕ: $Q_\theta(s, a) = \theta^\top \phi(s, a)$, where $\theta \in \mathbb{R}^d$ is a vector of weights.

The representation corresponds to a matrix $\Phi \in \mathbb{R}^{n \times d}$ whose rows are the vectors $\phi(s, a)$ for different state-action pairs (s, a). For clarity of presentation, we assume that Φ has full rank. A representation is *orthogonal* if $\Phi^\top \Xi \Phi = I$; these correspond to features which are normalized and uncorrelated. We write $\text{Span}(\Phi)$ to denote the subspace of value functions expressible using Φ, and denote Π the orthogonal projection operator onto $\text{Span}(\Phi)$, with closed form $\Pi = \Phi(\Phi^\top \Xi \Phi)^{-1} \Phi^\top \Xi$.

2.2. Temporal Difference Methods

TD fixed-point methods are a popular class of methods for approximate policy evaluation that attempt to find value functions that satisfy $Q = \Pi \mathcal{T}^\pi Q$ (Bradtke & Barto, 1996; Gordon, 1995; Maei et al., 2009; Dann et al., 2014). If $\Pi \mathcal{T}^\pi$ has a fixed-point, the solution is unique (Lagoudakis & Parr, 2003) and can be expressed as

$$\theta^*_{TD} = (\Phi^T \Xi (I - \gamma P^\pi) \Phi)^{-1} \Phi^T \Xi r.$$

We study TD(0), the canonical update rule to discover this fixed point. With a step size $\eta > 0$ and transitions sampled $(s, a, r, s', a') \sim \xi(s, a) P(s'|s, a) \pi(a'|s')$, TD(0) takes the update

$$\theta_{k+1} = \theta_k - \eta \nabla Q_{\theta_k}(s, a) \left(Q_{\theta_t}(s, a) - (r + \gamma Q_{\theta_t}(s', a')) \right).$$

In matrix form, this corresponds to an expected update over all state-action pairs:

$$\theta_{k+1} = \theta_k - \eta \left(\Phi^\top \Xi (I - \gamma P^\pi) \Phi \theta_k - \Phi^\top \Xi r \right). \quad (1)$$

With appropriately chosen decay of the step size, the stochastic update will converge if the expected update converges

(Benveniste et al., 1990; Tsitsiklis & Roy, 1996). However, these updates are not the gradient of any well-defined objective function except in special circumstances (Barnard, 1993; Ollivier, 2018), and hence do not inherit convergence properties from the classical optimization literature. The main aim of this paper is to provide conditions on the representation matrix Φ under which the update is convergent. We are especially interested in schemes that are convergent independent of the data distribution ξ.

We will characterize the stability of TD(0) and a representation through the spectrum of relevant matrices. For a matrix $A \in \mathbb{R}^{k\times k}$, the spectrum is the set of eigenvalues of A, written as $\text{Spec}(A) = \{\lambda_1, \ldots, \lambda_k\} \subset \mathbb{C}$. The spectral radius $\rho(A)$ denotes the maximum magnitude of eigenvalues. Stochastic transition matrices P^π satisfy $\rho(P^\pi) = 1$. We consider a potentially nonsymmetric matrix $A \in \mathbb{R}^{k\times k}$ to be positive definite if all non-zero vectors $x \in \mathbb{R}^k$ satisfy $\langle x, Ax\rangle > 0$.

2.3. Representation Learning

In reinforcement learning, a large class of methods have focused on constructing a representation Φ from the transition and reward functions, beginning perhaps with proto-value functions (Mahadevan & Maggioni, 2007). Involving P^π and r in the representation learning process is natural, since the value function Q^π is itself constructed from these two objects. As we shall later see, the stability criteria for these are also simple and coherent. Additionally, there is a large body of literature on the ease (or difficulty) with which these methods can be estimated from samples, and by proxy are amenable to gradient-descent schemes. Here we review the most common of these representation learning methods along with a few obvious extensions. Table 1 shows how their construction arises from different matrix operations on P^π and, in the case of the Krylov basis, of r.

Laplacian Representations: Proto-value functions (Mahadevan & Maggioni, 2007) capture the high-level structure of an environment, using the bottom eigenvectors of the normalized Laplacian of an undirected graph formed from environment transitions. This formalism extends to reversible Markov chains with on-policy data, but does not generalize to directional transitions, stochastic dynamics, and off-policy data. In the general setting, the Laplacian representation (Wu et al., 2018) uses the top eigenvectors of the symmetrized transition matrix (EigSymm) . We demonstrate in Section 4.3 that when data is off-policy, modifying the representation to omit eigenvectors whose eigenvalues exceed a threshold can provide strong stability guarantees.

Singular Vector Representations: Representations using singular vectors have been well-studied in representation learning for RL, because they are expressive and often yield strong performance guarantees. Fast Feature Selection (Behzadian et al., 2019) uses the top left singular vectors of the transition matrix as features. Similarly, Stachenfeld et al. (2014) and Machado et al. (2018) use the top left singular vectors of the successor representation (Dayan, 1993), a time-based representation which predicts future state visitations: $\Psi = (I - \gamma P^\pi)^{-1}$. We discover in Section 3.4 that the SVD objective of minimizing the norm of approximation error fails to preserve the spectral properties of transition matrices needed for stability, and can induce divergent behavior in TD(0) . In contrast, we show that decompositions constrained to preserve the spectrum of the transition matrix, such as the Schur decomposition, guarantee stability and performance.

REPRESENTATION	DECOMPOSITION
PROTO-VALUE FUNCTIONS [1]	EIG(P^π)
LAPLACIAN (EIGSYMM)	EIG($(P^\pi + \Xi^{-1}P^{\pi\top}\Xi)$)
SAFE EIGSYMM [2]	EIG($(P^\pi + \Xi^{-1}P^{\pi\top}\Xi)$)
SVD	SVD(P^π)
SVD OF SUCCESSOR REP.	SVD($(I - \gamma P^\pi)^{-1}$)
SCHUR	SCHUR(P^π)
KRYLOV BASIS	$\{r, P^\pi r, \ldots (P^\pi)^{d-1} r\}$
ORTHOG KRYLOV BASIS	ORTHOG($\mathcal{K}_d(P^\pi, r)$)

Table 1. Representation learning algorithms that learn features from the transition matrix and rewards. EIG is the spectral eigendecomposition, SVD the singular value decomposition, SCHUR the Schur decomposition, and ORTHOG an arbitrary orthogonal basis. [1] Only defined for reversible Markov chains with on-policy data. [2] Discards a partial set of features (see Section 4.3).

Reward-Informed Methods: If the reward structure of the problem is known apriori, a representation can focus its capacity on modelling future rewards and how they diffuse through the environment. Towards this goal, Petrik (2007) suggested the Krylov basis generated by P^π and r as features. Bellman Error Basis Functions (BEBFs) (Parr et al., 2007) iteratively builds a representation by adding the Bellman error for the best solution found so far as a new feature. Parr et al. (2008) show that under certain initial conditions for BEBFs, both representations span the Krylov subspace $\mathcal{K}_d(P^\pi, r)$ generated by rewards. Although no general guarantees exist for arbitrary rewards, we discover that when rewards are easily predictable, orthogonal representations that span this Krylov subspace have stability guarantees.

3. Stability Analysis of Arbitrary Representations

To begin, we study the stability of TD(0) given an arbitrary representation. For conciseness, we call TD(0) the algorithm whose expected update is described by equation 1; this is an algorithm which may or may not be off-policy (according to Ξ and P^π), and learns a linear approximation of the value function Q^π using features Φ. The following formalizes our

notion of stability.

Definition 3.1. *TD(0) is **stable** if there is a step-size $\eta > 0$ such that when taking updates according to equation 1 from any $\theta_0 \in \mathbb{R}^d$, we have $\lim_{k\to\infty} \theta_k = \theta^*_{TD}$.*

3.1. Learning Dynamics

For a sufficiently small step-size η, the discrete update of equation 1 behaves like the continuous-time dynamical system

$$\frac{\partial}{\partial t}(\theta_t - \theta^*_{TD}) = -A_\Phi(\theta_t - \theta^*_{TD}), \tag{2}$$

whose behaviour is driven by the *iteration matrix*

$$A_\Phi = \Phi^\top \Xi (I - \gamma P^\pi)\Phi.$$

Put another way, the learned parameters θ evolve approximately according to the linear dynamical system defined by the iteration matrix A_Φ. As might be expected, TD(0) is stable if this linear dynamical system is globally stable in the usual sense (Borkar & Meyn, 2000).

The iteration matrix – and as we shall see, the global stability of the linear dynamical system – depends on the data distribution, the representation, and, to a lesser extent, on the discount factor. It does not, however, depend on the reward function, which only affects the accuracy of the TD fixed-point solution θ^*_{TD}.

3.2. Stability Criteria

To understand the behaviour of TD(0), it is useful to contrast it with gradient descent on a weighted squared loss

$$\ell(\theta) = (\Phi\theta - \mathbf{y})^\top \Xi (\Phi\theta - \mathbf{y}),$$

where $\mathbf{y}$ is a vector of supervised targets. Gradient descent on $\ell(\theta)$ also corresponds to a linear dynamical system, albeit one whose iteration matrix is symmetric and positive definite. The behaviour of TD(0) is complicated by the fact that A_Φ is not guaranteed to be positive definite or symmetric, as the matrix ΞP^π itself is in general neither. In fact, the documented good behaviour of TD(0) arises in contexts where A_Φ itself is closer to a gradient descent iteration matrix: positive definite when the data distribution is on-policy (Tsitsiklis & Roy, 1996), and symmetric when the Markov chain described by P^π is reversible (Ollivier, 2018).

Following a well-known result from linear system theory (see e.g. Zadeh & Desoer, 2008), the asymptotic behavior of TD(0) more generally depends on the eigenvalues of the iteration matrix.

Proposition 3.1. *TD(0) is stable if and only if the eigenvalues of the implied iteration matrix A_Φ have positive real components, that is*

$$\mathit{Spec}(A_\Phi) \subset \mathbb{C}_+ := \{z : \mathit{Re}(z) > 0\}.$$

*We say that a particular choice of representation Φ is **stable** for (P^π, γ, Ξ) when A_Φ satisfies the above condition.*

Proof. See Appendix B for all proofs. □

Whenever the transition matrix, data distribution, and discount factor is evident, we will refer to Φ simply as a stable representation.

3.3. Effect of Subspace Parametrization

When measuring the approximation error that arises from a particular representation Φ, it suffices to consider the subspace spanned by the columns of Φ. It therefore makes no difference whether these columns are orthogonal (corresponding, informally speaking, to correlated features) or not. By contrast, we now show that the stability of the learning process does depend on how the linear subspace spanned by Φ is parametrized.

Recall that Φ is orthogonal if $\Phi^\top \Xi \Phi = I$. As it turns out, the stability of an orthogonal representation is determined by the *induced transition matrix* $\Pi P^\pi \Pi$, which describes how next-state features affect the TD(0) value estimates.

Proposition 3.2. *An orthogonal representation Φ is stable if and only if the real part of the eigenvalues of the induced transition matrix $\Pi P^\pi \Pi$ is bounded above, according to*

$$\mathit{Spec}(\Pi P^\pi \Pi) \subset \{z \in \mathbb{C} : \mathrm{Re}(z) < \tfrac{1}{\gamma}\}$$

In particular, Φ is stable if $\rho(\Pi P^\pi \Pi) < \frac{1}{\gamma}$.

Although the original transition matrix satisfies the spectral radius condition with $\rho(P^\pi) = 1$, the induced transition matrix can have eigenvalues beyond the stable region and lead to learning instability.

More generally, a representation Φ can be decomposed into an orthogonal basis and reparametrization $\Phi = \Phi' R$, where Φ' is an orthogonal representation spanning the same space as Φ and $R \in \mathbb{R}^{d\times d}$ is a reparametrization for Φ. The eigenvalues of the iteration matrix can be re-expressed as

$$\mathrm{Spec}(A_\Phi) = \mathrm{Spec}(R^\top A_{\Phi'} R) = \mathrm{Spec}(RR^\top A_{\Phi'}).$$

Despite spanning the same space, Φ and Φ' have iteration matrices with different spectra: $\mathrm{Spec}(A_\Phi) \neq \mathrm{Spec}(A_{\Phi'})$. As a result, the stability of Φ not only depends on the spectrum of $A_{\Phi'}$, but also how the reparametrization R shifts these eigenvalues. Put another way, Φ may be unstable even if its orthogonal equivalent Φ' is stable. The classical example of divergence given by Baird (1995) can be attributed to this phenomenon. In this example, the constructed representation expresses the same value functions as a stable tabular representation, but parametrizes the space in an different way and thus induces divergence.

3.4. Singular Vector Representations

The singular value decomposition is an appealing approach to representation learning: choosing vectors corresponding to large singular values guarantees, in a certain measure, low approximation error (Stachenfeld et al., 2014; Behzadian et al., 2019). Unfortunately, as now we show, doing so may be inimical to stability.

We denote Φ_{SVD} and Φ_{SR} the representations with the top d left singular vectors of P^π and Ψ as features. Recall that these vectors arise as part of a solution to a low-rank matrix approximation $\|A - \hat{A}\|_\Xi$. We write $\hat{P^\pi}, \hat{\Psi} \in \mathbb{R}^{n\times n}$ to denote the corresponding rank-d approximations.

Proposition 3.3 (SVD)**.** *The representation Φ_{SVD} is stable if and only if the low-rank approximation $\hat{P^\pi}$ satisfies*

$$\rho(\hat{P^\pi}) < \tfrac{1}{\gamma}.$$

Proposition 3.4 (Successor Representation)**.** *Recall that $Spec(\Psi) \subset \mathbb{C}_+$. The representation Φ_{SR} is stable if and only if the low-rank approximation $\hat{\Psi}$ satisfies*

$$Spec(\hat{\Psi}) \subset \mathbb{C}_+ \cup \{0\}.$$

Stability of a singular vector representation requires that the low-rank approximation maintain the spectral properties of the original matrix. This implies that such representations are *not stable* in general – the SVD low-rank approximation is chosen to minimize the norm of the error, and the spectrum of the approximation can deviate arbitrarily from the original matrix (Golub & van Loan, 2013). We note that the spectral conditions hold in the limit of almost-perfect approximation, but achieving this level of accuracy in practice may require an impractical number of additional features.

4. Representation Learning with Stability Guarantees

Our analysis of singular vector representations show that representations that optimize for alternative measures, such as approximation error, may lose properties of the transition matrix needed for stability. In this section, we study representations that are constrained, either in expressibility or in spectrum, to ensure stability.

4.1. Invariant Representations

We first consider representations whose induced transition matrix preserves the eigenvalues of the transition matrix to guarantee stability. These representations are closely linked to invariant subspaces of value functions that are closed under the transition dynamics of the policy.

Definition 4.1. *A representation Φ is P^π-**invariant** if its corresponding linear subspace is closed under P^π, that is*

$$Span(P^\pi\Phi) \subseteq Span(\Phi).$$

P^π-invariant subspaces are generated by the eigenspaces of P^π, and so invariant representations provide a natural way to reflect the geometry of the transition matrix. For these representations, we show that any eigenvalue of the induced transition matrix is also an eigenvalue of the transition matrix; this constraint ensures that invariant representations are always stable.

Theorem 4.1. *An orthogonal invariant representation Φ satisfies*

$$Spec(\Pi P^\pi \Pi) \subseteq Spec(P^\pi) \cup \{0\}$$

and is therefore stable.

Parr et al. (2008) studied the quality of the TD fixed-point solution on invariant subspaces, and found it to directly correlate with how well the subspace models reward. Our findings on stability emphasize the importance of their result – with invariant representations that can predict reward, good value functions not only exist, but are also reliably discovered by TD(0).

Although estimation of eigenvectors for a nonsymmetric matrix is numerically unstable, finding orthogonal bases for their eigenspaces can be done tractably, for example through the Schur decomposition.

Definition 4.2. *Let $A \in \mathbb{C}^{n\times n}$ be a complex matrix. A* Schur decomposition *of A, written $Schur(A)$, is URU^{-1}, where R is upper triangular and $U = [u_1, u_2, \ldots, u_n] \in \mathbb{C}^{n\times n}$ is orthogonal. For any k, $Span\{u_1, \ldots u_k\}$ is an A-invariant subspace.*

The Schur decomposition of P^π provides a sequence of vectors that span invariant subspaces, and can be constructed so that the first d basis vectors spans the top d-dimensional eigenspace of P^π. We define a representation using the first d Schur basis vectors to be the Schur representation.

When the transition matrix is reversible and data is on-policy, the Schur representation coincides with proto-value functions, and consequently also the successor representation (Machado et al., 2018). Unlike singular value representations, the Schur representation preserves the spectrum of the transition matrix at every step, and always guarantees stability.

Corollary 4.1.1. *The Schur representation is invariant and thus stable.*

A partial Schur basis can be constructed through orthogonal iteration, a generalized variant of power iteration.

Proposition 4.1 (Golub & van Loan (2013))**.** *Let $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$ be the ordered eigenvalues of P^π. If $|\lambda_d| > |\lambda_{d+1}|$ and $\Phi_0 \in \mathbb{C}^{n\times d}$, the sequence $\Phi_1, \Phi_2, \ldots$ generated via orthogonal iteration is*

$$\Phi_k = \textsc{Orthog}(Span(P^\pi\Phi_{k-1}))$$

where ORTHOG$(\cdot)$ *finds an orthogonal basis. As* $k \to \infty$, *Span*(Φ_k) *converges to the unique top eigenspace of* P^π.

In Section 5, we will see that the orthogonal iteration scheme can be approximated using a loss function and a target network (Mnih et al., 2015), and subsequently minimized with stochastic gradient descent, making it a potentially important tool for learning stable representations in practice.

4.2. Approximately Invariant Representations

In the previous section, we studied invariant representations, which are constrained to exactly preserve the eigenvalues of the transition matrix. We relax the notion of invariancy discuss a relaxation to approximate invariance, for which the spectrum of the induced matrix deviates from the transition matrix by a controlled amount, while still preserving stability. We find that approximate invariance leads to interesting implications for representations that span a Krylov subspace generated by rewards (Petrik, 2007; Parr et al., 2007).

Definition 4.3. *A representation is* ϵ*-invariant if*

$$\max_{v \in \mathit{Span}(\Phi)} \frac{\|\Pi P^\pi v - P^\pi v\|_\Xi}{\|v\|_\Xi} \leq \epsilon.$$

An approximately invariant representation spans a space in which the transition dynamics are not fully closed, but approximately so, as measured by the Ξ-norm. We provide a simple condition of when an ϵ-invariant representation is stable under assumptions of diagonalizability of the transition matrix. If P^π is diagonalizable with eigenbasis $A \in \mathbb{C}^{n \times n}$, the distance between the eigenvalues of the induced transition matrix $\Pi P^\pi \Pi$ and the original transition matrix P^π can be bounded by a function of a) ϵ, the degree of approximate invariance and b) the condition number of the eigenbasis $\kappa_\Xi(A) = \|A\|_\Xi \|A^{-1}\|_\Xi$ (Trefethen & Embree, 2005).

Theorem 4.2. *Let* Φ *be an orthogonal and* ϵ*-invariant representation for* (P^π, γ, Ξ) *. If* P^π *is diagonalizable with eigenbasis* A*, then* Φ *is stable if*

$$\epsilon < \frac{1-\gamma}{\gamma} \frac{1}{\kappa_\Xi(A)}.$$

This bound is quite stringent, especially for discount factors close to one and ill-conditioned eigenvector bases, but may be improved if the transition matrix has a special structure. For the general setting when the transition matrix is not diagonalizable, similar but more complicated bounds exist (Shi & Wei, 2012).

Approximately invariant representations are of particular interest when studying the Krylov subspace generated by rewards, $\mathcal{K}_d(P^\pi, r)$.

$$\mathcal{K}_d(P^\pi, r) = \text{Span}\{r, P^\pi r, \ldots, (P^\pi)^{d-1} v\}.$$

Representations that span this space admit a simple form of approximate invariancy.

Proposition 4.2. *A representation spanning* $\mathcal{K}_d(P^\pi, r)$ *is* ϵ*-invariant if*

$$\frac{\|\Pi P^\pi v - P^\pi v\|_\Xi}{\|v\|_\Xi} \leq \epsilon$$

Where $v = (I - \Pi_{d-1})(P^\pi)^{d-1} r$*, and* Π_{d-1} *is a projection onto the* $(d-1)$*-dimensional Krylov subspace* $\mathcal{K}_{d-1}(P^\pi, r)$.

Orthogonal representations for this Krylov subspace are approximately invariant if they can predict the reward at the $d+1$-th timestep well from the rewards attained in the first d timesteps. For rewards that diffuse through the environment rapidly and can be predicted easily, an orthogonal basis of the Krylov space generated by rewards is approximately invariant and thus stable. Challenging environments with sparse rewards and temporal separation however may require a prohibitively large Krylov space to guarantee stability. Note that there is an important distinction between orthogonal representations spanning a Krylov subspace and the Krylov basis itself: for most practical applications, rewards are highly correlated and because of the challenges of parametrization, the latter can be unstable.

4.3. Positive-Definite Representations

Invariant representations are stable because the spectrum of the projected transitions is constrained to closely mimic the eigenvalues of the transition matrix. What we call *positive definite representations* instead guarantee stability by constraining the set of expressible value functions to lie within a safe set. Positive definite representations are stable regardless of parametrization, unlike any family of representations discussed so far.

Definition 4.4. *The set of* positive-definite value functions $\mathcal{S}_{PD} \subset \mathbb{R}^n$ *is*

$$\mathcal{S}_{PD} = \{\, v \in \mathbb{R}^n \mid \langle v, P^\pi v \rangle_\Xi < \gamma^{-1} \|v\|_\Xi^2 \,\}.$$

Note that $\mathcal{S}_{PD}$ is not necessarily closed under addition. The two-state MDP presented by Tsitsiklis & Roy (1996) where TD(0) diverges can be interpreted through the lens of this set. For this example, the state representation only expresses value functions outside of $\mathcal{S}_{PD}$, which "grow" faster than γ^{-1}, and consequently leads to divergence. We focus on representations whose span falls within this set of safe value functions.

Definition 4.5. *We say that a representation is* ***positive-definite*** *if*

$$\mathit{Span}(\Phi) \subseteq \mathcal{S}_{PD}.$$

Note that a positive definite representation remains so under reparametrization, unlike the general case. In the special

case of on-policy data, $\mathcal{S}_{PD} = \mathbb{R}^n$ and all representations are positive-definite (Tsitsiklis & Roy, 1996).

Theorem 4.3. *A positive-definite representation Φ has a positive-definite iteration matrix A_Φ, and is thus stable.*

The Laplacian representation, which computes the spectral eigendecomposition of the symmetrized transition matrix

$$K := \frac{1}{2}\left(P^\pi + \Xi^{-1} P^{\pi\top} \Xi\right) = U \Lambda U^\top \Xi,$$

provides an interesting bifurcation of value functions into those that are positive-definite and those that are not. As a consequence of Theorem 4.3, a stable representation is obtained by using eigenvectors corresponding to eigenvalues smaller or equal to $\frac{1}{\gamma}$.

Proposition 4.3. *Let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of K, in decreasing order, and $u_1, \ldots, u_n$ the corresponding eigenvectors. Define d^* as the smallest integer such that $\lambda_{d^*} < \frac{1}{\gamma}$. For any $i \leq n - d^*$, the safe Laplacian representation Φ, defined as*

$$\Phi = [u_{d^*}, u_{d^*+1}, \ldots, u_{d^*+i}],$$

is positive-definite and stable.

While including eigenvectors for larger eigenvalues does not guarantee divergence, the basis $[u_1, \ldots, u_i]$ for $i < d^*$ *is* unstable (See appendix). When the data is on-policy, all eigenvalues of K are below the threshold $\frac{1}{\gamma}$, and the safe Laplacian corresponds exactly to the original representation.

We finish our discussion with a cautionary point. Although positive-definite representations admit amenable optimization properties, such as invariance to reparametrization and monotonic convergence, they can only express value functions that satisfy a growth condition. Under on-policy sampling this growth condition is nonrestrictive, but as the policy deviates from the data distribution, the expressiveness of positive-definite representations reduces greatly.

5. Experiments

We complement our theoretical results with an experimental evaluation, focusing on the following questions:

- How closely do the theoretical conditions we describe match stability requirements in practice?
- Can stable representations be learned using samples?
- Can they be learned using neural networks?

We conduct our study in the four-room domain (Sutton et al., 1999). We augment this domain with a task where the agent must reach the top right corner to receive a reward of +1 (Figure 1). The policy evaluation problem is to accurately estimate the value function of a near-optimal policy from data consisting of trajectories sampled by an uniform policy.

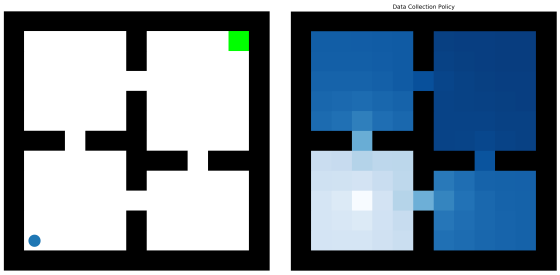

Figure 1. Left: The four room domain. Right: Data distribution

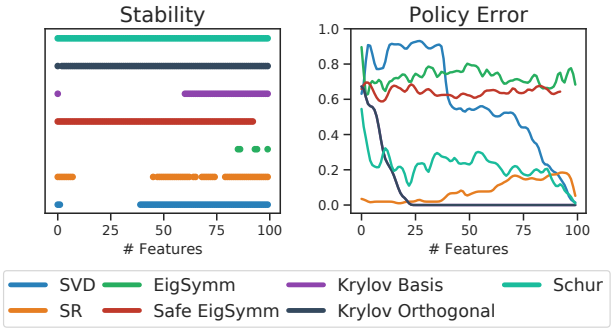

Figure 2. Stability (left) and approximation error (right) for different representation learning objectives. For stability, a marker is placed at d if the first d basis vectors forms a stable representation.

We are interested in the usefulness of the representation learning schemes summarized in Table 1 as a function of the number of features d that are used. We measure both the stability of the learned representation and its accuracy in estimating the greedy policy with respect to the fixed value function. We chose the latter measure as it is more informative than value approximation error when the number of features is small. See Appendix C for full details about the experimental setup.

Exact Representations: We first consider the quality of the representations in exact form, assuming access to the true transition matrix and reward function (Figure 2). We find that the general empirical profiles for stability match our theoretical characterizations. Singular vectors of the successor representation have low error but are unstable for most choices of small d. Although the Krylov basis of rewards and its orthogonalization both have the same estimation errors, they have drastically different stability profiles, confirming our analysis from Section 4.2. Amongst the proposed methods that consistently produce stable representations, the Schur basis admits low error and with enough features, is fully expressible. In contrast, the safe Laplacian representation takes an irrecoverable performance hit, as it discards the top eigenvectors of the symmetrized transition matrix that contain reward-relevant information.

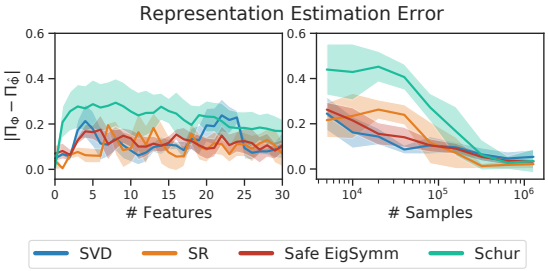

Figure 3. Learnability. We measure the difference between the true representation and one learned from an empirical transition matrix constructed from samples. Left: Error with 50000 transitions varying the number of features. Right: Error as the number of transitions varies when learning the first 10 features.

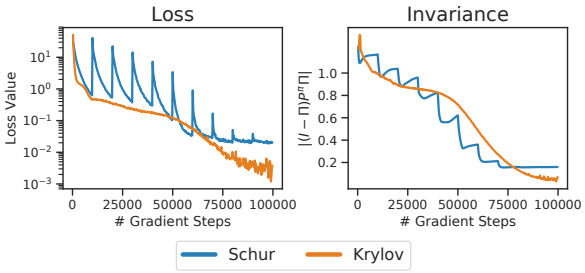

Figure 4. Learning to predict future rewards (Krylov) or future feature values (Schur) discovers approximately invariant stable representations.

Estimation with Samples: In practice, representations must be learned from finite data. To test the numerical robustness of the representation learning schemes, we construct an empirical transition matrix from a variable number of samples and learn a representation using this matrix.

We measure the difference between the subspaces spanned by the estimated and true representation (Figure 3). We find that estimating the Schur representation can be more challenging than the other methods, and requires an order of magnitude more data to accurately compute than representations for singular vectors and spectral decompositions. This is a well-known problem in numerical linear algebra, as eigenspaces for nonsymmetric matrices (SCHUR) are more sensitive to perturbation and estimation error than for eigenspaces of symmetric matrices (SPECTRAL, SVD). This implies a three-way tradeoff between stability, approximation error, and ease of estimation when choosing a representation for a general environment. The successor representation is unstable, the safe Laplacian is limited in its approximation power, and the Schur decomposition is harder to learn from samples. The orthogonal Krylov basis emerges as a strong method by these measures, but requires additional knowledge in the guise of the reward function.

Estimation with Neural Networks: In our final set of experiments, we show that the Schur representation and the orthogonal Krylov representation can be learned by neural networks by performing stochastic gradient descent on certain auxiliary objectives.

It has been noted previously that training a representation network with a final linear layer to predict features causes the neural network to learn a basis for the target features (Bellemare et al., 2019). A d-dimensional Krylov representation then can be learned by predicting reward values at the next d time-steps. Similarly, orthogonal iteration for learning the Schur representation (Proposition 3.2) can be approximated with a two-timescale algorithm that (a) at each step, predicts the feature values of a fixed target representation network at the *next time step* and (b) infrequently refreshes the target representation network with the current.

As our stability guarantees hold for orthogonal representations, the neural network must learn uncorrelated features, which can be enforced explicitly or with a penalty-based orthogonality loss (Wu et al., 2018). We fully describe the auxiliary objectives and provide implementation details in Appendix C.

Figure 4 demonstrates that these predictive losses can be optimized easily with neural networks and can learn stable approximately invariant representations. We note that this auxiliary task of predicting future latent states has been heuristically proposed before (François-Lavet et al., 2018; Gelada et al., 2019), as a way to improve approximation errors. Our results indicate that such auxiliary tasks may not only help reduce approximation error, but more importantly, can mitigate divergence in the learning process and provide for stable optimization.

6. Conclusion

We have presented an analysis of stability guarantees for value-function learning under various representation learning procedures. Our analysis provides conditions for stability of many algorithms that learn features from transitions, and demonstrates how representation learning procedures constrained to respect the geometry of the transition matrix can induce stability. We demonstrated that the Schur decomposition and orthogonal Krylov bases are rich representations that mitigate divergence in off-policy value function learning, and further showed that they can be learned using stochastic gradient descent on a loss function.

Our work provides formal evidence that representation learning can prevent divergence without sacrificing approximation quality. To carry our results to the full practical case, stability should be extended to the sequence of policies that are encountered during policy iteration. One should also consider the effects of learning value functions and representations concurrently, and the ensuing interactions in the representation. Our work suggests that studying stable representations in these contexts can be a promising avenue forward for the development of principled auxiliary tasks for stable deep reinforcement learning.

Acknowledgements

We would like to thank Nicolas Le Roux, Marlos C. Machado, Courtney Paquette, Fabian Pedregosa, Doina Precup, and Ahmed Touati for helpful discussions and contributions. We additionally thank Marlos C. Machado and Courtney Paquette for constructive feedback on an earlier manuscript.

References

Baird, L. C. Residual algorithms: Reinforcement learning with function approximation. In *ICML*, 1995.

Barnard, E. Temporal-difference methods and markov models. *IEEE Transactions on Systems, Man, and Cybernetics*, 1993.

Behzadian, B., Gharatappeh, S., and Petrik, M. Fast feature selection for linear value function approximation. In *ICAPS*, 2019.

Bellemare, M. G., Dabney, W., and Munos, R. A distributional perspective on reinforcement learning. In *ICML*, 2017.

Bellemare, M. G., Dabney, W., Dadashi, R., Taïga, A. A., Castro, P. S., Roux, N. L., Schuurmans, D., Lattimore, T., and Lyle, C. A geometric perspective on optimal representations for reinforcement learning. In *Advances in Neural Information Processing Systems*, 2019.

Benveniste, A., Priouret, P., and Métivier, M. *Adaptive Algorithms and Stochastic Approximations*. Springer-Verlag, Berlin, Heidelberg, 1990. ISBN 0387528946.

Bertsekas, D. P. Approximate policy iteration: a survey and some new methods. *Journal of Control Theory and Applications*, 9:310–335, 2011.

Bertsekas, D. P. Feature-based aggregation and deep reinforcement learning: a survey and some new implementations. *IEEE/CAA Journal of Automatica Sinica*, 6:1–31, 2018.

Bodnar, C., Li, A., Hausman, K., Pastor, P., and Kalakrishnan, M. Quantile QT-opt for risk-aware vision-based robotic grasping. *arXiv*, 2019.

Borkar, V. S. and Meyn, S. P. The o.d.e. method for convergence of stochastic approximation and reinforcement learning. *SIAM J. Control and Optimization*, 38:447–469, 2000.

Bradtke, S. J. and Barto, A. G. Linear least-squares algorithms for temporal difference learning. *Machine Learning*, 22:33–57, 1996.

Cabi, S., Colmenarejo, S. G., Novikov, A., Konyushkova, K., Reed, S., Jeong, R., Zolna, K., Aytar, Y., Budden, D., Vecerik, M., Sushkov, O., Barker, D., Scholz, J., Denil, M., de Freitas, N., and Wang, Z. A framework for data-driven robotics. *arXiv*, 2019.

Chung, W., Nath, S., Joseph, A. G., and White, M. Two-timescale networks for nonlinear value function approximation. In *International Conference on Learning Representations*, 2019.

Dalal, G., Szörényi, B., Thoppe, G., and Mannor, S. Finite sample analyses for td(0) with function approximation. In *AAAI*, 2017.

Dann, C., Neumann, G., and Peters, J. Policy evaluation with temporal differences: a survey and comparison. *J. Mach. Learn. Res.*, 15:809–883, 2014.

Dayan, P. Improving generalization for temporal difference learning: The successor representation. *Neural Computation*, 5:613–624, 1993.

François-Lavet, V., Bengio, Y., Precup, D., and Pineau, J. Combined reinforcement learning via abstract representations. *arXiv*, 2018.

Gelada, C., Kumar, S., Buckman, J., Nachum, O., and Bellemare, M. G. DeepMDP: Learning continuous latent space models for representation learning. In *Proceedings of the International Conference on Machine Learning*, 2019.

Golub, G. H. and van Loan, C. F. *Matrix Computations*. JHU Press, fourth edition, 2013. ISBN 1421407949 9781421407944. URL http://www.cs.cornell.edu/cv/GVL4/golubandvanloan.htm.

Gordon, G. J. Stable function approximation in dynamic programming. In *ICML*, 1995.

Jaderberg, M., Mnih, V., Czarnecki, W., Schaul, T., Leibo, J. Z., Silver, D., and Kavukcuoglu, K. Reinforcement learning with unsupervised auxiliary tasks. *ArXiv*, abs/1611.05397, 2016.

Lagoudakis, M. G. and Parr, R. Least-squares policy iteration. *J. Mach. Learn. Res.*, 4:1107–1149, 2003.

Levine, N., Zahavy, T., Mankowitz, D., Tamar, A., and Mannor, S. Shallow updates for deep reinforcement learning. In *Advances in Neural Information Processing Systems*, 2017.

Machado, M. C., Rosenbaum, C., Guo, X., Liu, M., Tesauro, G., and Campbell, M. Eigenoption discovery through the deep successor representation. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Bk8ZcAxR-.

Maei, H. R., Szepesvari, C., Bhatnagar, S., Precup, D., Silver, D., and Sutton, R. S. Convergent temporal-difference learning with arbitrary smooth function approximation. In *NIPS*, 2009.

Mahadevan, S. and Maggioni, M. Proto-value functions: A laplacian framework for learning representation and control in markov decision processes. *J. Mach. Learn. Res.*, 8:2169–2231, 2007.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540): 529–533, 2015.

Ollivier, Y. Approximate temporal difference learning is a gradient descent for reversible policies. *ArXiv*, abs/1805.00869, 2018.

Parr, R., Painter-Wakefield, C., Li, L., and Littman, M. L. Analyzing feature generation for value-function approximation. In *ICML '07*, 2007.

Parr, R., Li, L., Taylor, G., Painter-Wakefield, C., and Littman, M. L. An analysis of linear models, linear value-function approximation, and feature selection for reinforcement learning. In *ICML '08*, 2008.

Pathak, D., Agrawal, P., Efros, A. A., and Darrell, T. Curiosity-driven exploration by self-supervised prediction. *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 488–489, 2017.

Petrik, M. An analysis of laplacian methods for value function approximation in mdps. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pp. 2574–2579, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

Puterman, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., USA, 1st edition, 1994. ISBN 0471619779.

Shi, X. and Wei, Y. A sharp version of bauer–fike's theorem. *Journal of Computational and Applied Mathematics*, 236(13):3218 – 3227, 2012. ISSN 0377-0427. doi: https://doi.org/10.1016/j.cam.2012.02.021. URL http://www.sciencedirect.com/science/article/pii/S0377042712000787.

Stachenfeld, K. L., Botvinick, M., and Gershman, S. J. Design principles of the hippocampal cognitive map. In *Advances in Neural Information Processing Systems*, 2014.

Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*. MIT Press, 2nd edition, 2018.

Sutton, R. S., Precup, D., and Singh, S. P. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artif. Intell.*, 112:181–211, 1999.

Trefethen, L. N. and Embree, M. Spectra and pseudospectra : the behavior of nonnormal matrices and operators. 2005.

Tsitsiklis, J. N. and Roy, B. V. Analysis of temporal-diffference learning with function approximation. In *NIPS*, 1996.

van Hasselt, H., Doron, Y., Strub, F., Hessel, M., Sonnerat, N., and Modayil, J. Deep reinforcement learning and the deadly triad. *ArXiv*, abs/1812.02648, 2018.

Vecerik, M., Sushkov, O., Barker, D., Rothörl, T., Hester, T., and Scholz, J. A practical approach to insertion with variable socket position using deep reinforcement learning. 2019.

Wu, Y., Tucker, G., and Nachum, O. The laplacian in rl: Learning representations with efficient approximations. *ArXiv*, abs/1810.04586, 2018.

Yu, H. and Bertsekas, D. P. Basis function adaptation methods for cost approximation in mdp. In *Proceedings of the IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning*, 2009.

Zadeh, L. and Desoer, C. *Linear system theory: the state space approach*. Courier Dover Publications, 2008.