Towards a Better Understanding of Representation Dynamics under TD-learning

Yunhao Tang [1] **Rémi Munos** [1]

Abstract

TD-learning is a foundation reinforcement learning (RL) algorithm for value prediction. Critical to the accuracy of value predictions is the quality of state representations. In this work, we consider the question: how does end-to-end TD-learning impact the representation over time? Complementary to prior work, we provide a set of analysis that sheds further light on the representation dynamics under TD-learning. We first show that when the environments are reversible, end-to-end TD-learning strictly decreases the value approximation error over time. Under further assumptions on the environments, we can connect the representation dynamics with spectral decomposition over the transition matrix. This latter finding establishes fitting multiple value functions from randomly generated rewards as a useful auxiliary task for representation learning, as we empirically validate on both tabular and Atari game suites.

1. Introduction

Temporal difference (TD) learning is a foundational algorithm for predicting value functions in reinforcement learning (RL) (Sutton, 1988). In practice, computations of the value predictions depend on how the state is represented, a quantity formally known as state *representation*. In classic settings such as linear TD, the representations are often human designed and fixed throughout learning. In this case, the quality of the best possible value predictions depends critically on the quality of the fixed representations. Good representations should share useful information across states that entail accurate value predictions (see, e.g., (Tsitsiklis and Van Roy, 1996; Munos, 2003; Behzadian et al., 2019) for some example characterizations).

Any fixed representation is potentially sub-optimal, as it does not adapt to the underlying learning algorithm. To alleviate such issues, in practice, it is now common to perform gradient-based updates on the representations also with TD-learning. Recently, such an *end-to-end* approach to representation learning has led to much empirical success and is the core of many high-performing agents such as DQN (Mnih et al., 2013). A natural question ensues: can we characterize the representation learned by such end-to-end updates?

The answer to this question has been attempted by a number of prior work, including the study of the convergence of end-to-end TD-learning under the over-parameterized regimes, i.e., when the value functions are learned by very wide neural networks (Cai et al., 2019; Zhang et al., 2020; Agazzi and Lu, 2022; Sirignano and Spiliopoulos, 2022); the study of TD-learning dynamics under smooth homogeneous function approximation, e.g., with ReLU networks (Brandfonbrener and Bruna, 2019); the study of representation dynamics under TD-learning with restrictive assumptions on the weight parameter (Lyle et al., 2021). See Section 6 for an in-depth discussion about this paper's relation to prior work.

In this work, we provide a set of analysis complementary to prior work, which hopefully sheds light on how TD-learning impacts the evolution of representation over time. We consider the natural extension of the linear TD case (Sutton, 1988; Tsitsiklis and Van Roy, 1996), where the value function is parameterized in a bi-linear way

$$V = \Phi w.$$

Here, Φ is the representation, which is kept fixed in linear TD. We study the evolution of the representation under the general TD-learning updates. Our contributions consist in providing characterizations of the representation dynamics, through a few angles.

Improving value prediction accuracy over time. When assuming the Markov chain is reversible, we show that the value prediction error, as measured by the difference $\Phi w - V^\pi$, strictly decreases over time (Section 4). This solidifies the intuition that allowing updating representations during TD-learning should improve upon classic TD-learning where representations are fixed.

Spectral decomposition of the transition matrix . We show that when the transition matrix is symmetric and under

[1]Google DeepMind. Correspondence to: Yunhao Tang <robintyh@deepmind.com>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

certain conditions of reward functions, TD-learning learns provably useful representations (Section 5). Formally, the representation dynamics executes gradient-based spectral decomposition on the transition matrix.

Random value predictions learn useful representations. As a corollary of the previous results, we show that predicting multiple value functions generated via randomly sampled rewards is an auxiliary task that helps learn useful representations. We validate this theoretical insight with tabular and deep RL experiments over Atari game suites.

2. Background

Consider a Markov decision process (MDP) represented as the tuple $(\mathcal{X}, \mathcal{A}, p, p_R, \gamma)$ where $\mathcal{X}$ is a finite state space, $\mathcal{A}$ the finite action space, $p : \mathcal{X} \times \mathcal{A} \to \mathscr{P}(\mathcal{X})$ the transition kernel, $p_R : \mathcal{X} \times \mathcal{A} \to \mathbb{R}^h$ the reward kernel, and $\gamma \in [0, 1)$ the discount factor. In the traditional setting, the reward is scalar such that $h = 1$, though in general it can be extended to multiple dimensions $h \geq 1$ (Sutton et al., 2011). Let $\pi : \mathcal{X} \to \mathscr{P}(\mathcal{A})$ be a fixed policy. For convenience, let $P^\pi : \mathcal{X} \to \mathscr{P}(\mathcal{X})$ be the state transition kernel induced by the policy π such that $P^\pi(x, y) = \sum_a \pi(a|x)p(y|x, a)$. Denote the state-dependent reward function as $R^\pi \in \mathbb{R}^{|\mathcal{X}|\times h}$ such that $R^\pi(x) = \sum_a r(x, a)\pi(a|x) \in \mathbb{R}^h$. Throughout, we will focus on the Markov reward process under P^π and R^π as this simplifies discussions.

When the context is clear, we overload the notation and let x be an one-hot encoding of the state too. A state representation $\phi_x \in \mathbb{R}^k$ is defined by a mapping from the state space $\mathcal{X}$ to the k-dimensional Euclidean space. To characterize representations, we consider a matrix $\Phi \in \mathbb{R}^{|\mathcal{X}|\times k}$ from which the representation for state x can be calculated as $\phi_x := \Phi^T x \in \mathbb{R}^k$. Throughout, we assume $k \leq |\mathcal{X}|$ where $|\mathcal{X}|$ is the cardinal of $\mathcal{X}$, as this tends to be the case in practice. Good representations should entail sharing information between states, and facilitate downstream tasks such as policy evaluation or control. In classic TD-learning settings, representations are fixed, whereas in practice, representations are also shaped by incremental updates, as we detail below.

2.1. TD-learning with linear function approximations

Given the representation $\phi_x \in \mathbb{R}^k$ at state x, TD-learning with linear function approximation parameterizes a linear function with weight $w \in \mathbb{R}^{k\times h}$, such that the prediction $\phi_x^T w$ approximates the value function $V^\pi := (I - \gamma P^\pi)^{-1} R^\pi \in \mathbb{R}^{|\mathcal{X}|\times h}$. In most applications, there is a single reward $h = 1$; here, we consider the most general case $h \geq 1$ as this helps facilitate ensuing discussions. For any $1 \leq i \leq h$, we can understand the i-th column of the weight vector w_i as approximating the value function $V_i^\pi \in \mathbb{R}^{|\mathcal{X}|}$ derived from the i-th column of the reward matrix $R_i^\pi \in \mathbb{R}^{|\mathcal{X}|}$. The aim of TD-learning is to adjust the weight parameter w such that the approximation is accurate. We start with the classic linear TD-learning setting where the representation Φ is fixed.

Given a state x_t and its sampled next state $x_{t+1} \sim P^\pi(\cdot|x_t)$, TD-learning constructs the bootstrapped back-up target $R^\pi(x_t) + \gamma\phi_{x_{t+1}}^T w$ and seeks to minimize the squared prediction error

$$\left\| R^\pi(x_t) + \gamma\phi_{x_{t+1}}^T w - \phi_{x_t}^T w \right\|_2^2.$$

TD-learning is defined through the *semi-gradient* update, which can be understood as gradient descent on the modified squared prediction error with a stop gradient operation on the back-up target,

$$L(\Phi, w) := \left\| \text{sg}\left(R^\pi(x_t) + \gamma\phi_{x_{t+1}}^T w \right) - \phi_{x_t}^T w \right\|_2^2. \quad (1)$$

Here we omit the loss function's dependency on the sampled transition for simplicity. Then, the weight is updated as

$$w_{t+1} = w_t - \eta_w \partial_{w_t} L(\Phi, w_t), \quad (2)$$

with learning rate $\eta_w \geq 0$. Throughout, we assume the state x_t are drawn from the stationary distribution $d^\pi \in \mathbb{R}^{|\mathcal{X}|}$ of the Markov matrix P^π. We let $D^\pi \in \mathbb{R}^{|\mathcal{X}|\times|\mathcal{X}|}$ denote the diagonal matrix constructed from d^π. Assuming Φ is of rank k and $d^\pi > 0$, it has been proved that under mild conditions on the learning rate and the representation matrix, linear TD-learning converges to the unique fixed point (Tsitsiklis and Van Roy, 1996)

$$w_\Phi^* := \left(\Phi^T D^\pi (I - \gamma P^\pi)\Phi\right)^{-1} \Phi^T D^\pi R^\pi. \quad (3)$$

The primary approach taken by the seminal work of (Tsitsiklis and Van Roy, 1996) is to understand linear TD-learning through the continuous time behavior of the expected linear TD updates, characterized by an ordinary differential equation (ODE). Now, we overload the notation and let w_t be the weight parameter indexed by continuous time $t \geq 0$. The continuous time behavior of linear TD is characterized by the following ODE

$$\frac{dw_t}{dt} = \eta_w \cdot \Phi^T D^\pi \left(R^\pi - (I - \gamma P^\pi)\Phi w_t\right). \quad (4)$$

Here, $\eta_w > 0$ is the learning rate. We can verify that w_Φ^* is the unique fixed point to Equation (4).

3. End-to-end linear TD-learning: jointly updating weight and representation

We now provide further background on a natural extension of linear TD-learning to the case where the representations

are updated as well, a case we call *deep* TD-learning (Lyle et al., 2021). In end-to-end linear TD, both representation Φ and weight w are updated by semi-gradient descent on the squared prediction error in Equation (1). The joint update on Φ_t and w_t is

$$\begin{aligned} w_{t+1} &= w_t - \eta_w \partial_{w_t} L(\Phi_t, w_t), \\ \Phi_{t+1} &= \Phi_t - \eta_\Phi \partial_{\Phi_t} L(\Phi_t, w_t) \end{aligned} \tag{5}$$

with learning rates $\eta_w, \eta_\Phi \geq 0$. If we interpret the combination of representation and weight parameter $\theta = (\Phi, w)$ as a whole, the end-to-end linear TD update in Equation (5), combined with the original linear TD update in Equation (2) can be understood as semi-gradient descent on the loss function $L(\Phi, w)$ with respect to θ. Though end-to-end linear TD-learning brings us closer to practical implementations (e.g., DQN (Mnih et al., 2013)), there is no general guarantee on the behavior of the joint system (Φ_t, w_t).

We aim to understand the behavior of end-to-end linear TD by characterizing the behavior of its continous time system. The continuous time ODE of the end-to-end linear TD updates in Equation (5) can be formally stated as follows.

Lemma 1. The ODE to the end-to-end linear TD update in Equation (5) is

$$\begin{aligned} \frac{dw_t}{dt} &= \eta_w \cdot \Phi_t^T D^\pi \left(R^\pi - (I - \gamma P^\pi)\, \Phi_t w_t\right), \\ \frac{d\Phi_t}{dt} &= \eta_\Phi \cdot D^\pi \left(R^\pi - (I - \gamma P^\pi)\, \Phi_t w_t\right) w_t^T. \end{aligned} \tag{6}$$

A critical difference between the linear TD and end-to-end linear TD is demonstrated through the ODE systems in Equation (4) and Equation (6): the linear TD induces a *linear* ODE in the variable w_t, whereas the end-to-end linear TD is a coupled non-linear ODE in the joint variable (Φ_t, w_t). This hints at the difficulty in characterizing the learning dynamics of the end-to-end linear TD as alluded to earlier.

Remarks on the learning rates. The linear TD can be understood as a special case when $\eta_\Phi = 0$. In practical implementations, it is more common to update both set of parameters with the same learning rate $\eta_w = \eta_\Phi > 0$. An interesting case for theoretical analysis is when the representation is updated at a much slower pace than the weight vector, i.e., $\eta_\Phi \ll \eta_w$. We study such a limiting case in Section 5. Below, we consider the general case where both learning rates assume positive finite values.

4. Value function approximation error of end-to-end linear TD

A primary quantity of interest for TD-learning is the value approximation error $\Phi w - V^\pi$. In linear TD-learning, given the fixed Φ, the approximation error is in general finite even if w converges to the fixed point w_Φ^*. In end-to-end linear TD-learning, where the representation matrix Φ_t is adapted over time based on the semi-gradient, intuitively we should expect the approximation error to decrease over time. Below, we make a formal characterization of such an intuition.

As a measure of the approximation accuracy, we define the following weighted value approximation error $E(\Phi, w)$,

$$E \coloneqq \frac{1}{2}\mathrm{Tr}\left(\left(\Phi w - V^\pi\right)^T D^\pi (I - \gamma P^\pi)\left(\Phi w - V^\pi\right)\right),$$

where the function $\mathrm{Tr}(M) = \sum_i M_{ii}$ returns the trace of a squared matrix M. The error is effectively the sum of weighted norms for the columns of $\Phi w - V^\pi \in \mathbb{R}^{|\mathcal{X}| \times h}$ under the key matrix

$$D^\pi (I - \gamma P^\pi) \in \mathbb{R}^{|\mathcal{X}| \times |\mathcal{X}|}.$$

The key matrix is positive definite (PD), see, e.g., (Sutton et al., 2016) for a detailed proof. As a result, the weighted error enjoys a few useful properties.

Lemma 2. The weighted error $E(\Phi, w)$ is always non-negative and evaluates to zero if and only if $\Phi w = V^\pi$.

As a first important conclusion of this work, we show that when the MDP is reversible, the error function strictly decreases over time, as long as (Φ_t, w_t) has not converged to a critical point of the ODE dynamics. Intriguingly, this is because under the reversibility assumption, the end-to-end linear TD dynamics turns out to be gradient descent on the error function.

Theorem 3. Assume the Markov chain is reversible, i..e, $D^\pi P^\pi = (P^\pi)^T D^\pi$, then the end-to-end linear TD dynamics in Equation (6) is effectively gradient descent on the error function $E(\Phi_t, w_t)$, i.e.,

$$\frac{dw_t}{dt} = -\eta_w \cdot \partial_{w_t} E(\Phi_t, w_t), \quad \frac{d\Phi_t}{dt} = -\eta_\Phi \cdot \partial_{\Phi_t} E(\Phi_t, w_t).$$

As a result, the weighted value approximation error is non-increasing $dE(\Phi_t, w_t)/dt \leq 0$. If (Φ_t, w_t) is not at a critical point of the learning dynamics, i.e. when $\frac{d\Phi_t}{dt} \neq 0$ or $\frac{dw_t}{dt} \neq 0$, then $dE(\Phi_t, w_t)/dt$ is strictly negative.

Proof. We provide a proof sketch here. The reversibility assumption establishes that the key matrix $D^\pi(I - \gamma P^\pi)$ is symmetric. Under this symmetric condition, with a few lines of algebraic manipulations, we can show that indeed the updates to w_t and Φ_t follow negative gradient on the weighted error $E(\Phi_t, w_t)$. As a result

$$\frac{d}{dt} E(\Phi_t, w_t) = -\left(\frac{1}{\eta_\Phi}\left\|\frac{d\Phi_t}{dt}\right\|_2^2 + \frac{1}{\eta_w}\left\|\frac{dw_t}{dt}\right\|_2^2\right),$$

where $\|\cdot\|_2$ is the L_2 norm. This concludes the proof. □

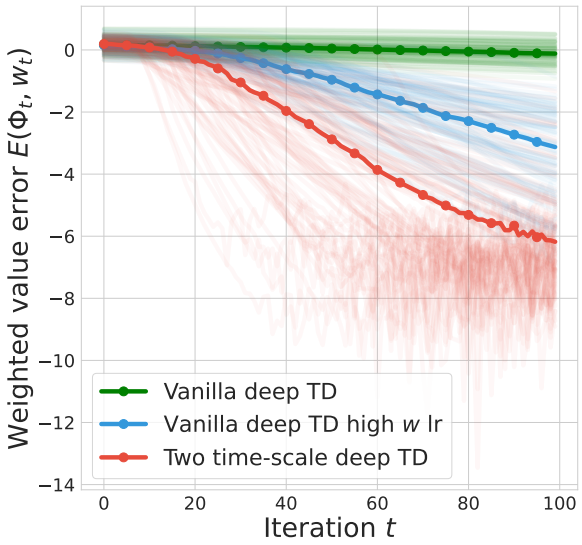

Figure 1. The evolution of the weighted value approximation error $E(\Phi_t, w_t)$ under the end-to-end linear TD dynamics. The solid curves show the median result over 100 randomly generated MDPs. We compare three cases: (1) vanilla end-to-end linear TD with $\eta_\Phi = \eta_w = 1$ (Equation (6)); (2) vanilla end-to-end linear TD with $\eta_\Phi = 1, \eta_w = 10$; (3) two time-scale end-to-end linear TD (Equation (9)). Across all dynamics, the weighted error takes a decaying trend though such decay is not necessarily monotonic for randomly sampled MDPs. Fixing the number of iterations, the two time-scale dynamics achieves the fastest rate of decay compared to vanilla end-to-end linear TD.

We expand on the result a bit more. The second term in the time derivative $\left\|\frac{dw_t}{dt}\right\|_2^2$ can be attributed to the classic linear TD update, whereas the first term $\left\|\frac{d\Phi_t}{dt}\right\|_2^2$ is derived from the updates on the representation matrix. Such a *local improvement* property is indicative of the usefulness of end-to-end linear TD in practice, similar to how policy gradient updates provide local value improvements in policy optimization. The rate of the improvement in $E(\Phi_t, w_t)$ is also proportional to the magnitude of the semi-gradient updates to Φ_t and w_t.

Critical points of end-to-end linear TD. Theorem 3 implies the end-to-end linear TD dynamics should converge to a critical point, i.e., points at which $\dot{\Phi}_t = 0$ and $\frac{dw_t}{dt} = 0$, which we discuss below. Solving for $\frac{dw_t}{dt} = 0$, we see that given any Φ, the weight parameter must be at the TD fixed point $w = w^*_\Phi$. Then when solving for $\dot{\Phi} = 0$, we obtain the following characterization for the critical points.

Lemma 4. Assume the matrix $\Phi_t^T D^\pi (I - \gamma P^\pi)\Phi_t$ is invertible for all time, then the set of critical points $\mathcal{C}$ of the end-to-end linear TD dynamics (Equation (6)) are pairs of $(\Phi, w) \in \mathcal{C}$ such that $w = w^*_\Phi$, and Φ satisfies the following condition,

$$D^\pi R^\pi (D^\pi R^\pi)^T \Phi \in \text{span}\left(D^\pi (I - \gamma P^\pi)\Phi\right). \quad (7)$$

Here, span(A) denotes the vector subspace spanned by the columns of A. The notation $A \in$ span(B) means that each column of matrix A is in the subspace span(B).

We note that in general, it is challenging to deliver a more concise and intuitive description of the critical points beyond Equation (7), however, with further assumptions we can provide more meaningful characterizations on the representations at critical points. See Section 5 for further discussions.

Reversibility assumption. The reversibility assumption on the Markov chain is a sufficient condition in showing that the weighed error $E(\Phi_t, w_t)$ strictly decays over time (Theorem 3), similar to the linear TD case (Ollivier, 2018). Lately, Brandfonbrener and Bruna (2019) also showed the convergence of non-linear TD dynamics when the MDP is reversible enough. All such prior results allude to the important impact of such an assumption. Nevertheless, it is a very strong assumption in general. We complement the theoretical results here with numerical simulations of the weighted error $E(\Phi_t, w_t)$ along the end-to-end linear TD dynamics, over general MDPs that can violate the reversibility assumption.

In Fig. 1, we show the evolution of $E(\Phi_t, w_t)$ over time under three different dynamics: (1) vanilla end-to-end linear TD with $\eta_\Phi = \eta_w = 1$ (Equation (6)); (2) vanilla end-to-end linear TD with $\eta_\Phi = 1, \eta_w = 10$; (3) two time-scale end-to-end linear TD (Equation (9), see Section 5 for details). For each dynamics, we generate 100 random MDPs that very likely violate the reversibility assumption. In all cases, the error takes a decaying trend though the decrease is not necessarily monotonic in general. This empirically shows that to certain degree, the general trend of *decreasing value error* is robust to the violation of the assumption. Furthermore, we see that when the weight parameter is updated at a higher learning rate than the representation matrix, the error decays at a faster rate. The fastest rate seems to be obtained at the extreme when at each iteration, the weight parameter is set at the TD fixed point $w_t = w^*_{\Phi_t}$ under the limiting two time-scale dynamics (see Section 5). See Appendix D for further experimental details.

Convergence of the error and representation. Since the error function $E(\Phi_t, w_t)$ is strictly monotonic over time, it converges to an asymptotic value as $t \to \infty$. An important question is whether the error converges to the lowest possible value over the set of critical points $\lim_{t\to\infty} E(\Phi_t, w_t) = \inf_{(\Phi,w)\in\mathcal{C}} E(\Phi, w)$. Answering such questions requires a refined analysis of the dynamics, which we leave to future

work. A related question is what does the representation Φ_t converge to, if it converges at all. We provide some discussions on this question in the next section.

5. Characterizing representations of end-to-end linear TD

Thus far we have characterized the behavior of the end-to-end linear TD dynamics (in Equation (6)) via the value approximation error $E(\Phi, w)$. Such an error characterizes the information that the aggregate prediction Φw contains. In applications such as policy evaluation, where value prediction is the ultimate objective, such a characterization seems to suffice.

In practice, it is also of interest to understand the information that the representation Φ contains by itself. Consider a motivating example (taken from (Bellemare et al., 2019)) where $k = h = 1$, if we take the representation to be the value function $\Phi = V^\pi$ and $w = 1$, the prediction error is zero. However, such a representation is specialized to reward function R^π and does not capture general information about the transition matrix P^π.

5.1. Warming up: two time-scale end-to-end linear TD

To facilitate the discussion, we consider a specialized case of the end-to-end linear TD dynamics, where the weight parameter w_t is updated at a much faster time pace than Φ_t. Numerically, this can be emulated by the end-to-end linear TD updates in Equation (5) with $\eta_\Phi \ll \eta_w$ as mentioned earlier. Here, we consider the extreme case, the *two time-scale* dynamics where at at any iteration t, the weight parameter $w_t = w^*_{\Phi_t}$ is the optimal weight adapted to representation Φ_t.

$$w_t = w^*_{\Phi_t},\ \Phi_{t+1} = \Phi_t - \eta_\Phi \partial_{\Phi_t} L(\Phi_t, w_t) \tag{8}$$

We call the above dynamics the end-to-end linear TD-learning with two time-scales. The continuous time ODE system is as follows.

Lemma 5. The ODE to the two time-scale dynamics in Equation (9) is

$$\frac{d\Phi_t}{dt} = \eta_\Phi \cdot D^\pi \left(R^\pi - (I - \gamma P^\pi)\Phi_t w^*_{\Phi_t}\right)(w^*_{\Phi_t})^T, \tag{9}$$

where recall that $w^*_{\Phi_t}$ is the linear TD fixed point to representation Φ_t specified in Equation (3).

Importantly, note that the two time-scale ODE dynamics cannot be recovered exactly from the ODE to the vanilla end-to-end linear TD Equation (6) with finite values of η_w and η_Φ. Nevertheless, as a corollary to Theorem 3, we can still show that under the reversibility assumption, the two time-scale dynamics also decreases the weighted value approximation error.

Corollary 6. Assume the Markov chain is reversible, then under the two time-scale learning dynamics in Equation (9), the weighted value approximation error is non-increasing $\frac{d}{dt}E(\Phi_t, w_t) \leq 0$. If Φ_t is not at a critical point of the learning dynamics, then $\frac{d}{dt}E(\Phi_t, w_t)$ is strictly negative.

5.2. Non-collapse representation dynamics

One primary motivation to consider the two time-scale dynamics is that the representation matrix Φ_t enjoys the following important property: throughout the dynamics, the covariance matrix $\Phi_t^T\Phi_t$ is a constant matrix over time.

Lemma 7. Under the two time-scale dynamics in Equation (9), the covariance matrix $\Phi_t^T\Phi_t \in \mathbb{R}^{k\times k}$ is a constant matrix over time.

Lemma 7 implies that the representation matrix Φ_t maintains its representational capacity over time. Denoting the i-th column of Φ_t as $\Phi_{t,i}$ for $1 \leq i \leq k$, we can visualize the evolution of Φ_t as *rotations* in the representation space: the relative angle between $\Phi_{t,i}$ and $\Phi_{t,j}$, as well as their respective lengths, are preserved over time. For example, if the k columns of Φ_t are initialized to be orthornormal $\Phi_0^T\Phi_0 = I_{k\times k}$, then they stay orthonormal throughout the dynamics. This excludes situations where all k columns converge to the same direction (see, e.g., dynamics in (Lyle et al., 2021) for examples), in which case the learned representation is not ideal.

5.3. When does end-to-end linear TD learn useful representations

In classic linear TD, the representations are fixed throughout the weight update. Examples of good *fixed* representations include top eigen or singular vector decompositions of the transition matrix P^π (Mahadevan, 2005; Machado et al., 2018; Behzadian et al., 2019; Ren et al., 2022), some of which can lead to provably low value approximation errors (Behzadian et al., 2019).

Now, we seek to identify cases where end-to-end linear TD entails learning such useful representations. To measure the information contained in Φ_t about the transition matrix P^π, we define the trace objective

$$f(\Phi_t) \coloneqq \mathrm{Tr}\left(\Phi_t^T(I - \gamma P^\pi)^{-1}\Phi_t\right). \tag{10}$$

The resolvant matrix $(I - \gamma P^\pi)^{-1} = \sum_{t=0}^\infty (\gamma P^\pi)^t$ reflects the long term discounted transition under P^π. To see why the above trace objective provides a helpful measure, note that when P^π is symmetric, we can relate it to the data covariance matrix in PCA (Abdi and Williams, 2010). Indeed, if we constrain inputs to f to be the matrix U concatenating any of k distinct eigenvectors $(u_i)_{i=1}^k$ of $(I - \gamma P^\pi)^{-1}$, then $f(U) = \sum_{i=1}^k \lambda_i$ where λ_i is the eigenvalue of u_i. As a

result, solving the constrained optimization problem

$$\max_{\Phi} f(\Phi), \text{ s.t. } \Phi^T\Phi = I_{k\times k} \tag{11}$$

amounts to finding the top k eigenvectors of $(I-\gamma P^\pi)^{-1}$.

Can we show that the end-to-end linear TD dynamics makes progress towards maximizing $f(\Phi)$? Unfortunately, even under the restrictive setting that P^π is symmetric, there is no guarantee that $f(\Phi_t)$ is maximized following the end-to-end linear TD dynamics. An probably intuitive explanation is that since R^π comes into play in the TD updates, it may prevent Φ_t from learning just about the transition matrix (see numerical examples in Section 5.4). However, by imposing further conditions on the reward function, we show below that Φ_t learns to capture the spectral information of P^π.

Theorem 8. Assume the outer product of the reward function is an identity matrix $R^\pi(R^\pi)^T = I_{|\mathcal{X}|\times|\mathcal{X}|}$ and assume P^π is symmetric. Then under the two time-scale dynamics in Equation (9), the trace objective is non-increasing $df/dt \geq 0$ over time. If $\frac{d\Phi_t}{dt} \neq 0$, then $df/dt > 0$.

We will discuss the implication of $R^\pi(R^\pi)^T = I$ in Section 5.4. Before, that, we note a critical observation: by Lemma 7 and Theorem 8, we see that the two time-scale end-to-end linear TD dynamics carries out gradient-based optimization to solve the constrained k-PCA problem (Equation (11)). A key requirement is to satisfy the orthonormal constraints. Fortunately, as Lemma 7 suggests, if we initialize Φ_0 to be orthonormal, the constraints $\Phi_t^T\Phi_t = I$ is implicitly satisfied throughout the dynamics.

Critical points span eigenspaces of P^π. When assumptions in Theorem 8 are imposed, we can strengthen our characterizations of the critical points of the end-to-end linear TD dynamics, by relating them to eigen subspaces of P^π. Here, we limit our attention to *non-trivial* critical points that satisfy the constraints $\Phi^T\Phi = \Phi_0^T\Phi_0$ as other critical points cannot be arrived at from the learning dynamics (Lemma 7).

Corollary 9. When assumptions in Theorem 8 are satisfied, and further, assume $\Phi_0^T D^\pi(I-\gamma P^\pi)\Phi_0$ is invertible. A representation Φ is a non-trivial critical point of the two time-scale dynamics (Equation (9)) if and only if it spans an invariant subspace of P^π, i.e., $P^\pi\Phi \in \text{span}(\Phi)$.

5.4. Fitting values with randomly sampled rewards

Recall that $R^\pi \in \mathbb{R}^{|\mathcal{X}|\times h}$ can be understood as the concatenation of h different reward functions, and we let $R_i^\pi \in \mathbb{R}^{|\mathcal{X}|}$ denote its i-th column for $1 \leq i \leq h$. An especially interesting case is when entries of each one of the h such reward columns are sampled i.i.d.

Lemma 10. (**Randomly sampled rewards**) When the each entry of R_i^π is sampled i.i.d. from any zero-mean distribution with finite variance $\sigma^2/\sqrt{h}$ for some constant $\sigma > 0$, for $1 \leq i \leq h$. Then as the number of reward columns $h \to \infty$, we have $R^\pi(R^\pi)^T \to \sigma^2 I_{|\mathcal{X}|\times|\mathcal{X}|}$ almost surely.

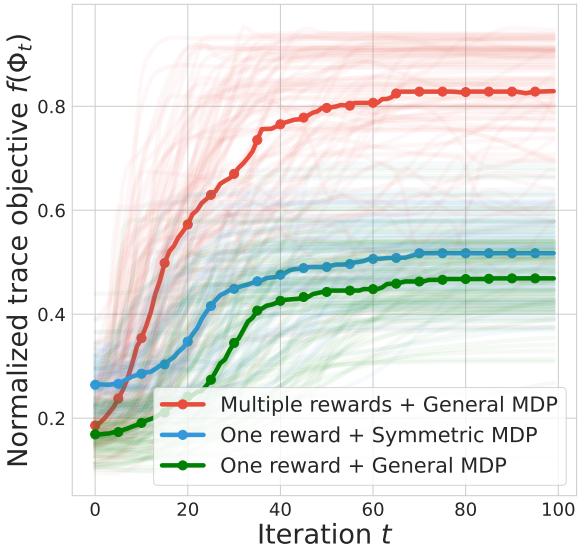

Figure 2. The normalized trace objective $f(\Phi_t)$ under the end-to-end linear TD dynamics. We generate 100 random MDPs and solid curves show the median results. We consider three scenarios: (1) multiple reward functions $h = 5$ with general P^π; (2) single reward $h = 1$ with symmetric P^π; (3) single reward with general P^π. The trace objective takes an upward trend though the improvement is not necessarily monotonic for randomly sampled MDPs. When predicting multiple random rewards, the representations seem to capture more spectral information about P^π as measured by the trace objective.

Now, it is useful to recall Theorem 3 which showed that the weighted value errors decay over time under the end-to-end linear TD dynamics. In light of this previous result, we can interpret Theorem 8 and its connections to randomly sampled rewards as follows: if starting from a representation Φ_t, one can accurately fit value functions V_i^π generated by multiple random rewards R_i^π simultaneously, then Φ_t must capture spectral information about the transition matrix P^π.

Numerical experiments on randomly sampled MDPs. In practice, the symmetry assumption in Theorem 8 is likely to be violated. In Fig. 2 we evaluate the trace objective $f(\Phi_t)$ under the two time-scale dynamics with randomly generated MDPs. We compare a few cases: (1) multiple reward functions $h = 5$ with general P^π; (2) single reward $h = 1$ with symmetric P^π; (3) single reward with general P^π. The results suggest that the upward trend of the trace objective is fairly persistent even in asymmetric MDPs, implying that the dynamics entails Φ_t to capture useful information about P^π in general. However, the improvement in f is not monotonic in general.

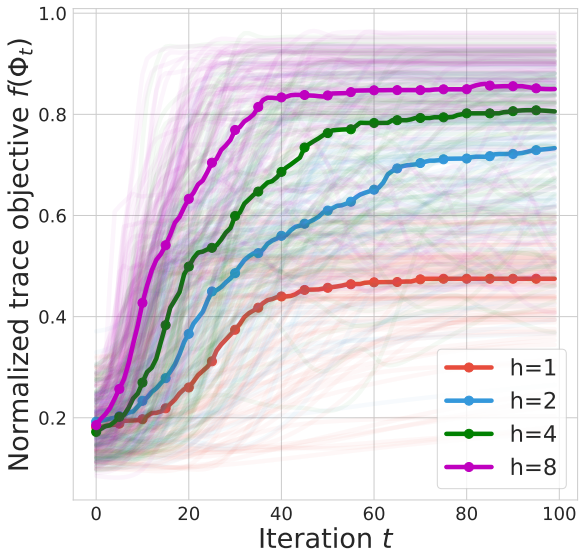

Figure 3. The normalized trace objective $f(\Phi_t)$ under the end-to-end linear TD dynamics. We generate 100 random MDPs with general P^π and solid curves show the median results. We study the effect of the number of random rewards h, with $h \in \{1, 2, 4, 8\}$. As h increases, the median trace objective $f(\Phi_t)$ improves both at higher rates and achieves higher asymptotic values, indicating that the representations converge to the top k eigensubspace of P^π. In this case, the state space $|\mathcal{X}| = 30$ and $h = 4$ seems to suffice for learning very good representations.

Another critical observation is that by fitting more random reward functions at the same time (red $h = 5$ vs. green $h = 1$), the dynamics seemingly captures more spectral information about the transition. The result also implies that in practice, we probably do not need a very large number h of rewards, see Section 7 for an ablation experiment.

How many reward functions are enough? The above arguments imply that we might need a large number of reward functions for the condition $R^\pi(R^\pi)^T = I$ to hold. In practice, a key empirical question is how many random reward functions suffice for the end-to-end linear TD dynamics to learn good representations. We carry out an ablation study in Fig. 3 where we consider $h \in \{1, 2, 4, 8\}$. As h increases, the trace objective $f(\Phi_t)$ both improves at a higher average rate, and achieves higher asymptotic values. In our tabular experiments, we consider $k = 2$ dimensions for the representations. Intriguingly, $h = 8$ seems to suffice to learning good representations (when the normalized trace objective ≥ 0.8; note that if P^π is symmetric, the maximum normalized trace objective is 1), which is noticeably smaller than the number of states $|\mathcal{X}| = 30$. Though the conclusion certainly depends on the specific MDPs, the result suggests in practice, potentially a relatively small number of random rewards is enough to induce good representations.

Connections to auxiliary tasks in prior work. The idea of fitting value functions of randomly generated rewards bears close connections to a number of prior approach for building auxiliary tasks in deep RL, such as random cumulant predictions (Dabney et al., 2021; Zheng et al., 2021). Random cumulant predictions consist in learning h value functions starting from a single representation Φ_t, with different value *heads*. In our terminology, such value heads can be understood as separate weight parameter $w_i \in \mathbb{R}^k, 1 \leq i \leq h$, each dedicated to learning value function $V_i^\pi \in \mathbb{R}^{|\mathcal{X}|}$ with randomly generated reward R_i^π. This is mathematically equivalent to our matrix notation with $w \in \mathbb{R}^{k \times h}$ throughout the paper.

6. Discussion on related work

A more comprehensive discussion is in Appendix C.

From linear TD to end-to-end linear TD. Much of the prior work has focused on the linear TD setup (Sutton and Barto, 1998; Tsitsiklis and Van Roy, 1996), where the representations are assumed fixed throughout learning. Closely related to our work is (Lyle et al., 2021) where they proposed to understand the learning dynamics of end-to-end linear TD through its corresponding ODE system. Since such an ODE system is highly non-linear, it is more challenging to provide generic characterizations without restrictive assumptions. Lyle et al. (2021) bypasses the non-linearity issue by essentially assuming a fixed weight parameter $w_t \equiv w$. In light of Equation (6), they study the following dynamics

$$\frac{d\Phi_t}{dt} = \eta_\Phi \cdot D^\pi \left(R^\pi - (I - \gamma P^\pi)\Phi_t w\right) w^T$$

which effectively reduces to a linear system in Φ_t and is more amenable to analysis.

Our work differs in a few important aspects. Firstly, our analysis adheres strictly to the vanilla end-to-end linear TD (Equation (6)) or two-time scale end-to-end linear TD (Equation (9)), without imposing the constant weight assumption as in (Lyle et al., 2019). Our analysis is also slightly more general, as it consists in constructing a few scalar functions that characterize the dynamics. This is more applicable to general ODEs where obtaining exact solutions is not tractable.

TD with non-linear function approximations. Along the line of research on TD-dynamics with non-linear function approximation, a closely related work is (Brandfonbrener and Bruna, 2019) where they established the convergence behavior of TD-learning with smooth homogeneous functions. A key requirement underlying their result is that the MDP is sufficiently reversible, which echos the assumption we make in Theorem 3. However, under their framework

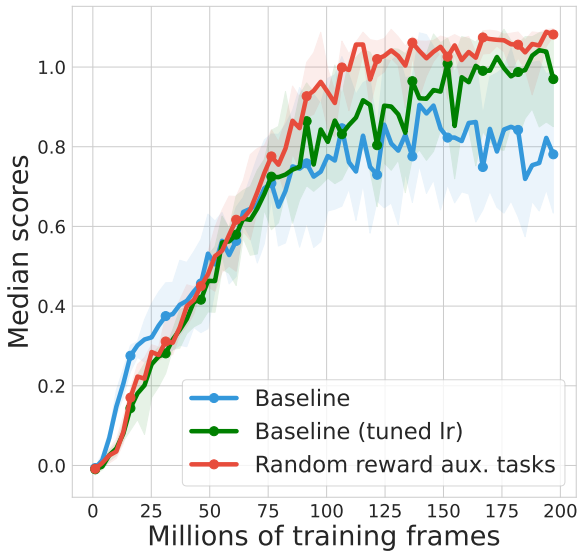

Figure 4. Comparison of training curves of median human normalized scores across 15 Atari games. We compare the baseline DQN, DQN with tuned learning rate and DQN with random value predictions as auxiliary tasks. The shaded areas show the 95% bootstrapped confidence intervals averaged over 3 seeds. The random value prediction tasks seem to provide marginal benefits over the tuned DQN. See Appendix D for further details.

there is no clear notion of representation as defined in the bi-linear case.

Self-predictive learning. Motivated by value-based TD-learning, self-predictive learning directly employs the notion of bootstrapping to the representation space, which has produced a number of empirically successful implementations (Guo et al., 2020; Schwarzer et al., 2021; Guo et al., 2020). Recently, Tang et al. (2022) proposes to understand the behavior of self-predictive learning through its corresponding ODE. From their discussion, we note that two time-scale dynamics is a generic way to enforcing optimization constraints. Our results in Section 5 build on this and show a similar characterization for the end-to-end linear TD.

Two time-scale learning. Levine et al. (2017) proposed a combination of LSTD updates for the linear weights on top of DQN representations, which are shaped by gradient descents. In continuous time, this is effectively the two time-scale dynamics (Eqn (9)). While their discussion is empirically motivated, we focus on the representation learning aspect of such a learning dynamics.

Auxiliary tasks. In deep RL literature, it has been empirically observed that certain extra training objectives, normally referred to as *auxiliary tasks* (see, e.g., (Beattie et al., 2016; Jaderberg et al., 2016; Bellemare et al., 2019; Fedus et al., 2019; Dabney et al., 2021; Zheng et al., 2021)). Though the conventional wisdom is that such auxiliary tasks are useful in shaping representations, it is generally difficult to characterize exactly what representations are induced, despite some recent theoretical efforts (Lyle et al., 2021). Our results formally demonstrate the potential benefits of random reward predictions when combined with TD-learning.

7. Experiments

We validate the potential usefulness of fitting randomly generated value functions as an auxiliary task in deep RL settings. We use DQN (Mnih et al., 2013) as a baseline, and generate random reward functions $R_i^\pi(x, a)$ via outputs of randomly initialized networks, following the practice of (Dabney et al., 2021). The common deep RL equivalent of the representation ϕ_x is the output of the torso network (usually a convnet). For the auxiliary task, we fit multiple value function by applying different head network (usually MLPs) on top of the common output ϕ_x. Throughout, we update all network parameters with the same learning rate. We have ablated on increasing the learning rate for head networks, so that the algorithm is more in line with the two time-scale dynamics; however, we found that in practice, this tends to provide the torso network with less learning signal (as much is taken care of by the head network) and usually impedes performance. Understanding the theory-practice gap here would be an important future direction.

Fig. 5 shows the result for DQN, DQN with tuned learning rate, and DQN with random value function auxiliary task. Overall, we see that the auxiliary task provides marginal improvement on the tuned DQN baseline, hinting at the potential usefulness of the auxiliary objective for representation learning. See Appendix D for further experimental details.

8. Conclusion

We have provided a number of theoretical analysis on the representation dynamics under end-to-end end-to-end linear TD dynamics. Under the reversibility assumption, we have showed that the representation evolves such that the value approximation error decreases over time. With further assumptions on the reward function, we establish that the representation dynamics carries out spectral decomposition on the transition matrix, leading to provably useful representations. This further implies that fitting random value functions is a principled auxiliary task for representation learning.

Our work opens up a few interesting directions for further investigation, such as how to relax the reversibility assumptions, how to combine the end-to-end linear TD framework with nonlinear functions and extensions to the control case.

References

Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C Courville, and Marc Bellemare. Deep reinforcement learning at the edge of the statistical precipice. *Advances in neural information processing systems*, 34: 29304–29320, 2021.

Andrea Agazzi and Jianfeng Lu. Temporal-difference learning with nonlinear function approximation: lazy training and mean field regimes. In *Mathematical and Scientific Machine Learning*, pages 37–74. PMLR, 2022.

Charles Beattie, Joel Z Leibo, Denis Teplyashin, Tom Ward, Marcus Wainwright, Heinrich Küttler, Andrew Lefrancq, Simon Green, Víctor Valdés, Amir Sadik, et al. Deepmind lab. *arXiv preprint arXiv:1612.03801*, 2016.

Bahram Behzadian, Soheil Gharatappeh, and Marek Petrik. Fast feature selection for linear value function approximation. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 29, pages 601–609, 2019.

Marc Bellemare, Will Dabney, Robert Dadashi, Adrien Ali Taiga, Pablo Samuel Castro, Nicolas Le Roux, Dale Schuurmans, Tor Lattimore, and Clare Lyle. A geometric perspective on optimal representations for reinforcement learning. *Advances in neural information processing systems*, 32, 2019.

Marc G. Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.

David Brandfonbrener and Joan Bruna. Geometric insights into the convergence of nonlinear td learning. *arXiv preprint arXiv:1905.12185*, 2019.

Qi Cai, Zhuoran Yang, Jason D Lee, and Zhaoran Wang. Neural temporal-difference and q-learning provably converge to global optima. *arXiv preprint arXiv:1905.10027*, 2019.

Will Dabney, André Barreto, Mark Rowland, Robert Dadashi, John Quan, Marc G Bellemare, and David Silver. The value-improvement path: Towards better representations for reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7160–7168, 2021.

William Fedus, Carles Gelada, Yoshua Bengio, Marc G Bellemare, and Hugo Larochelle. Hyperbolic discounting and learning over multiple horizons. *arXiv preprint arXiv:1902.06865*, 2019.

Zhaohan Daniel Guo, Bernardo Avila Pires, Bilal Piot, Jean-Bastien Grill, Florent Altché, Rémi Munos, and Mohammad Gheshlaghi Azar. Bootstrap latent-predictive representations for multitask reinforcement learning. In *International Conference on Machine Learning*, pages 3875–3886. PMLR, 2020.

Max Jaderberg, Volodymyr Mnih, Wojciech Marian Czarnecki, Tom Schaul, Joel Z Leibo, David Silver, and Koray Kavukcuoglu. Reinforcement learning with unsupervised auxiliary tasks. *arXiv preprint arXiv:1611.05397*, 2016.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*, 2015.

Nir Levine, Tom Zahavy, Daniel J Mankowitz, Aviv Tamar, and Shie Mannor. Shallow updates for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 30, 2017.

Clare Lyle, Marc G. Bellemare, and Pablo Samuel Castro. A comparative analysis of expected and distributional reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4504–4511, 2019.

Clare Lyle, Mark Rowland, Georg Ostrovski, and Will Dabney. On the effect of auxiliary tasks on representation dynamics. In *International Conference on Artificial Intelligence and Statistics*, pages 1–9. PMLR, 2021.

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

Hamid Maei, Csaba Szepesvari, Shalabh Bhatnagar, Doina Precup, David Silver, and Richard S Sutton. Convergent temporal-difference learning with arbitrary smooth function approximation. *Advances in neural information processing systems*, 22, 2009.

Sridhar Mahadevan. Proto-value functions: Developmental reinforcement learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 553–560, 2005.

Francisco S Melo, Sean P Meyn, and M Isabel Ribeiro. An analysis of reinforcement learning with function approximation. In *Proceedings of the 25th international conference on Machine learning*, pages 664–671, 2008.

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

Rémi Munos. Error bounds for approximate policy iteration. In *ICML*, volume 3, pages 560–567. Citeseer, 2003.

Yann Ollivier. Approximate temporal difference learning is a gradient descent for reversible policies. *arXiv preprint arXiv:1805.00869*, 2018.

Tongzheng Ren, Tianjun Zhang, Lisa Lee, Joseph E Gonzalez, Dale Schuurmans, and Bo Dai. Spectral decomposition representation for reinforcement learning. *arXiv preprint arXiv:2208.09515*, 2022.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.

Max Schwarzer, Ankesh Anand, Rishab Goel, R Devon Hjelm, Aaron Courville, and Philip Bachman. Data-efficient reinforcement learning with self-predictive representations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=uCQfPZwRaUu.

Justin Sirignano and Konstantinos Spiliopoulos. Asymptotics of reinforcement learning with neural networks. *Stochastic Systems*, 12(1):2–29, 2022.

Richard S. Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3(1):9–44, 1988.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

Richard S Sutton, Joseph Modayil, Michael Delp, Thomas Degris, Patrick M Pilarski, Adam White, and Doina Precup. Horde: A scalable real-time architecture for learning knowledge from unsupervised sensorimotor interaction. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 761–768, 2011.

Richard S Sutton, A Rupam Mahmood, and Martha White. An emphatic approach to the problem of off-policy temporal-difference learning. *The Journal of Machine Learning Research*, 17(1):2603–2631, 2016.

Yunhao Tang, Zhaohan Daniel Guo, Pierre Harvey Richemond, Bernardo Ávila Pires, Yash Chandak, Rémi Munos, Mark Rowland, Mohammad Gheshlaghi Azar, Charline Le Lan, Clare Lyle, et al. Understanding self-predictive learning for reinforcement learning. *arXiv preprint arXiv:2212.03319*, 2022.

John Tsitsiklis and Benjamin Van Roy. Analysis of temporal-diffference learning with function approximation. *Advances in neural information processing systems*, 9, 1996.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

Yufeng Zhang, Qi Cai, Zhuoran Yang, Yongxin Chen, and Zhaoran Wang. Can temporal-difference and q-learning learn representation? a mean-field theory. *arXiv preprint arXiv:2006.04761*, 2020.

Zeyu Zheng, Vivek Veeriah, Risto Vuorio, Richard L Lewis, and Satinder Singh. Learning state representations from random deep action-conditional predictions. *Advances in Neural Information Processing Systems*, 34:23679–23691, 2021.

APPENDICES: Towards a Better Understanding of Representation Dynamics under TD-learning

A. Derivation of continuous time ODEs for end-to-end linear TD

Recall the squared prediction error in Equation (1),

$$L(\Phi, w) := \left\| \text{sg}\left(R^\pi(x_t) + \gamma \phi_{x_{t+1}}^t w\right) - \phi_{x_t}^T w \right\|_2^2.$$

We take an expectation over the state distribution $x_t \sim d^\pi, x_{t+1} \sim P^\pi(\cdot|x_t)$, the expected loss function can be expressed as follows

$$\mathbb{E}[L] = \sum_x d^\pi(x) \left(\text{sg}\left(R^\pi(x) + \gamma \left(P^\pi \Phi w\right)(x)\right) - \left(\Phi w\right)(x)\right)^2,$$

where $(\Phi w)(x)$ and $(P^\pi \Phi w)(x)$ denote the x-th coordinate of the corresponding vector $\Phi w \in \mathbb{R}^{|\mathcal{X}|}$ and $P^\pi \Phi w \in \mathbb{R}^{|\mathcal{X}|}$. The expected semi-gradient for Φ and w can be computed as

$$\partial_w \mathbb{E}[L] = -\Phi^T D^\pi \left(R^\pi - \left(I - \gamma P^\pi\right)\Phi w\right), \partial_\Phi \mathbb{E}[L] = -D^\pi \left(R^\pi - \left(I - \gamma P^\pi\right)\Phi w\right) w^T. \tag{12}$$

Recall that end-to-end linear TD is defined as follows

$$\frac{dw_t}{dt} = -\eta_w \cdot \partial_w \mathbb{E}[L], \ \frac{d\Phi_t}{dt} = -\eta_\Phi \cdot \partial_w \mathbb{E}[L].$$

Plugging into the formula in Equation (12), we arrive at the ODE system in Lemma 1.

Now, to derive the two time-scale ODE dynamics, we just need to replace the generic weight vector w_t in the update to Φ_t above by the fixed point $w^*_{\Phi_t}$. This gives rise to the ODE in Equation (9).

B. Proof

Lemma 2. The weighted error $E(\Phi, w)$ is always non-negative and evaluates to zero if and only if $\Phi w = V^\pi$.

Proof. Let $v_{t,i}$ denote the i-th column of $\Phi w - V^\pi$ for $1 \le i \le h$. We have

$$\begin{aligned} E(\Phi, w) &= \text{Tr}\left(\left(\Phi w - V^\pi\right)^T D^\pi (I - \gamma P^\pi)\left(\Phi w - V^\pi\right)\right) \\ &= \sum_{i=1}^h v_{t,i}^T D^\pi (I - \gamma P^\pi) v_{t,i}. \end{aligned}$$

Since $D^\pi(I - \gamma P^\pi)$ is PD (Sutton et al., 2016), we have each of the i-th term above is non-negative and is only zero when $v_{t,i} = 0$. This concludes the result. □

Theorem 3. Assume the Markov chain is reversible, i..e, $D^\pi P^\pi = (P^\pi)^T D^\pi$, then the end-to-end linear TD dynamics in Equation (6) is effectively gradient descent on the error function $E(\Phi_t, w_t)$, i.e.,

$$\frac{dw_t}{dt} = -\eta_w \cdot \partial_{w_t} E(\Phi_t, w_t), \ \frac{d\Phi_t}{dt} = -\eta_\Phi \cdot \partial_{\Phi_t} E(\Phi_t, w_t).$$

As a result, the weighted value approximation error is non-increasing $dE(\Phi_t, w_t)/dt \le 0$. If (Φ_t, w_t) is not at a critical point of the learning dynamics, i.e. when $\frac{d\Phi_t}{dt} \ne 0$ or $\frac{dw_t}{dt} \ne 0$, then $dE(\Phi_t, w_t)/dt$ is strictly negative.

Proof. For notational simplicity, let $A = D^\pi(I - \gamma P^\pi)$. Under the reversibility assumption, A is symmetric $A^T = A$. We also define $v_t := \Phi_t w_t - V^\pi$ and denote $v_{t,i}$ as its i-th column. Define the matrix $M = \left(\Phi w - V^\pi\right)^T A \left(\Phi w - V^\pi\right)$, then note that its (i, j)-thn component is $v_{t,i}^T A v_{t,j}$. As a result, the weighted error rewrites as

$$E(\Phi_t, w_t) = \sum_{i=1}^h M_{ii} = \sum_{i=1}^h \underbrace{\frac{1}{2} v_{t,i}^T A v_{t,i}}_{=: M_{ii}}.$$

Let $w_{t,i}$ be the i-th column of matrix w_t. Recall by definition of the end-to-end linear TD dynamics, we have

$$\frac{d\Phi_t}{dt} = -\eta_\Phi \cdot A\left(\Phi_t w_t - V^\pi\right) w_t^T = -\eta_w \cdot A \sum_{i=1}^{h} v_{t,i} w_{t,i}^T$$
$$\frac{dw_{t,i}}{dt} = -\Phi_t^T A v_{t,i}.$$

In order to derive $\partial_{\Phi_t} E(\Phi_t, w_t)$ and $\partial_{w_t} E(\Phi_t, w_t)$, note a few useful facts as follows,

$$\partial_{\Phi_t} M_{ii} = \frac{1}{2}(A + A^T) v_{t,i} w_{t,i}^T = A v_{t,i} w_{t,i}^T,$$
$$\partial_{w_t} M_{ii} = \frac{1}{2}\Phi^T (A + A^T) v_{t,i} = \Phi^T A v_{t,i}.$$

As a result, we can verify

$$\frac{d\Phi_t}{dt} = -\eta_\Phi \sum_{i=1}^{h} \partial_{\Phi_t} M_{ii} = -\eta_\Phi \partial_{\Phi_t} E(\Phi_t, w_t)$$
$$\frac{dw_t}{dt} = -\eta_w \sum_{i=1}^{h} \partial_{w_t} M_{ii} = -\eta_w \partial_{w_t} E(\Phi_t, w_t),$$

where (a) comes from the fact that $A = A^T$. Now with chain rule, we have

$$\frac{dE(\Phi_t, w_t)}{dt} = \text{Tr}\left(\left(\partial_{\Phi_t} E(\Phi_t, w_t)\right)^T \frac{d\Phi_t}{dt}\right) + \text{Tr}\left(\left(\partial_{w_t} E(\Phi_t, w_t)\right)^T \frac{dw_t}{dt}\right)$$
$$= -\left(\frac{1}{\eta_\Phi}\left\|\frac{d\Phi_t}{dt}\right\|_2^2 + \frac{1}{\eta_w}\left\|\frac{dw_t}{dt}\right\|_2^2\right) \leq 0,$$

which is strictly negative if (Φ_t, w_t) is not at a critical point. The proof is hence concluded. □

Lemma 4. Assume the matrix $\Phi_t^T D^\pi (I - \gamma P^\pi)\Phi_t$ is invertible for all time, then the set of critical points $\mathcal{C}$ of the end-to-end linear TD dynamics (Equation (6)) are pairs of $(\Phi, w) \in \mathcal{C}$ such that $w = w_\Phi^*$, and Φ satisfies the following condition,

$$D^\pi R^\pi (D^\pi R^\pi)^T \Phi \in \text{span}\left(D^\pi (I - \gamma P^\pi)\Phi\right). \tag{7}$$

Here, span(A) denotes the vector subspace spanned by the columns of A. The notation $A \in$ span(B) means that each column of matrix A is in the subspace span(B).

Proof. For convenience, define $A := D^\pi(I - \gamma P^\pi)$. We set $\frac{dw_t}{dt} = 0$ and $\frac{d\Phi_t}{dt} = 0$. It is straightforward to see that $\frac{dw_t}{dt} = 0$ effectively allows us to derive the TD fixed point

$$w_t = \left(\Phi_t^T A \Phi_t\right)^{-1} \Phi_t^T A V^\pi =: w_{\Phi_t}^*.$$

Note that for the above expression, we have used the assumption that $\Phi_t^T A \Phi_t$ is invertible. Then, solving for $\frac{d\Phi_t}{dt} = 0$ and noting that $w_t = w_{\Phi_t}^*$, we have

$$A\Phi_t \left(\Phi_t^T A \Phi_t\right)^{-1} \Phi_t^T A V^\pi (V^\pi)^T A^T \Phi (\Phi_t^T A^T \Phi_t)^{-1} = A V^\pi (V^\pi)^T A^T \Phi (\Phi_t^T A^T \Phi_t)^{-1}.$$

Cancelling out the invertible matrix at the end, we have

$$\underbrace{A\Phi_t \left(\Phi_t^T A \Phi_t\right)^{-1} \Phi_t^T}_{M} A V^\pi (V^\pi)^T A^T \Phi = A V^\pi (V^\pi)^T A^T \Phi. \tag{13}$$

Now, consider the projection matrix M, which satisfies $M^2 = M$. We exploit the following useful property of the projection matrix

$$Mx = \arg\min_{y \in \text{span}(A\Phi)} \|x - y\|^2_{(A^T)^{-1}},$$

where $\|x-y\|^2_{(A^{-1})^T} := (x-y)(A^T)^{-1}(x-y)$ is the squared weighted norm of $x - y$ under the PD matrix $(A^T)^{-1}$ (we show that this matrix is indeed PD towards the end of the proof). Applying the result of Lemma 11 to each column of the vector $AV^\pi(V^\pi)^T A^T \Phi$, we have Equation (13) is equivalent to below

$$AV^\pi(V^\pi)^T A^T \Phi \in \text{span}\,(A\Phi)\,. \tag{14}$$

From Equation (14), we derive a set of equivalent conditions.

$$\begin{aligned} AV^\pi(V^\pi)^T A^T \Phi \in \text{span}\,(A\Phi) &\iff_{(a)} \exists B \in \mathbb{R}^{k\times k},\ AV^\pi(V^\pi)^T A^T \Phi = A\Phi B \\ &\iff_{(b)} (I - \gamma P^\pi)^{-1} R^\pi (R^\pi)^T (D^\pi)^T \Phi = \Phi B \\ &\iff_{(c)} D^\pi R^\pi (R^\pi)^T (D^\pi)^T \Phi = A\Phi B \\ &\iff_{(d)} D^\pi R^\pi (R^\pi)^T (D^\pi)^T \Phi \in \text{span}(A\Phi). \end{aligned}$$

Here, (a) follows from the definition of span$(A\Phi)$; (b) follows by canceling A on both sides of the equation; (c) follows from some straightforward algebraic manipulations; (d) follows from the definition of span$(A\Phi)$. This concludes the proof.

PD of matrix $(A^T)^{-1}$. Recall that A is PD, and by definition this means $x^T A x \geq 0$ and is only zero for $x = 0$. Given any x, we want to show

$$x^T (A^T)^{-1} x \geq 0$$

and is only zero for $x = 0$. To see this, for any fixed vector x we define $y = A^{-1}x$. Since A is PD, A^{-1} is invertible and $x = 0$ if and only if $y = 0$. Now, we have

$$x^T (A^T)^{-1} x = y^T A y,$$

which is non-negative for all $y \neq 0$ and is only zero when $y = 0$, in which case $x = 0$. This shows the matrix $(A^T)^{-1}$, though asymmetric in general, is also PD.

□

Lemma 11. For any PD matrix D and matrix A with compatible shapes, define the projection matrix $M := A(A^T D A)^{-1} A^T D$. Let v be a vector. The following two results are equivalent: (1) $Mv = v$; (2) $v \in \text{span}(A)$.

Proof. By definition of the projection matrix M, we can write

$$Mv = \arg\min_{y \in \text{span}(A)} \|v - y\|^2_D\,.$$

Starting with condition (1), we know that v is a feasible solution to the optimization problem above. This means $v \in \text{span}(A)$, which is condition (2). Starting with condition (2), we know that v is a feasible solution to the optimization problem that defines the projection. It is straightforward to verify that v is also the unique optimal solution (unique because D is PD). Hence, we have $Mv = v$. □

Corollary 6. Assume the Markov chain is reversible, then under the two time-scale learning dynamics in Equation (9), the weighted value approximation error is non-increasing $\frac{d}{dt}E(\Phi_t, w_t) \leq 0$. If Φ_t is not at a critical point of the learning dynamics, then $\frac{d}{dt}E(\Phi_t, w_t)$ is strictly negative.

Proof. Using chain rule as in the proof of Theorem 3, for any generic argument (Φ_t, w_t) to the weighted loss function, we have

$$\frac{dE(\Phi_t, w_t)}{dt} = \text{Tr}\left(\left(\partial_{\Phi_t} E(\Phi_t, w_t)\right)^T \frac{d\Phi_t}{dt} \right) + \text{Tr}\left(\left(\partial_{w_t} E(\Phi_t, w_t)\right)^T \frac{dw_t}{dt} \right).$$

Recall that by definition the TD fixed point $w^*_{\Phi_t}$ satisfies the equation

$$\Phi_t^T D^\pi (I - \gamma P^\pi)(\Phi_t w^*_{\Phi_t} - V^\pi) = 0.$$

Meanwhile, the proof in Theorem 3 has showed

$$\partial_{w_t} E(\Phi_t, w_t) = -\Phi_t^T D^\pi (I - \gamma P^\pi)(\Phi_t w_t - V^\pi).$$

Combining the above two results, we have $\partial_{w^*_{\Phi_t}} E(\Phi_t, w^*_{\Phi_t}) = 0$. Further, using the result from Theorem 3 we have $\frac{d\Phi_t}{dt} = -\eta_\Phi \cdot \partial_{\Phi_t} E(\Phi_t, w^*_\Phi)$. This implies the following

$$\frac{dE(\Phi_t, w^*_{\Phi_t})}{dt} = -\frac{1}{\eta_\Phi} \left\| \frac{d\Phi_t}{dt} \right\|_2^2,$$

which concludes the proof. □

Lemma 7. Under the two time-scale dynamics in Equation (9), the covariance matrix $\Phi_t^T \Phi_t \in \mathbb{R}^{k \times k}$ is a constant matrix over time.

Proof. Our result follows closely the proof technique in (Tang et al., 2022). Under the two time-scale TD dynamics, the weight $w_t = w^*_{\Phi_t}$ is the fixed point solution given the representation Φ_t. Let $A_t = \Phi_t w_t \in \mathbb{R}^{|\mathcal{X}| \times h}$ be the matrix product. The chain rule combined with the first-order optimality condition on w_t (which defines the fixed point) implies

$$\partial_{w_t} \mathbb{E}\left[L(\Phi_t, w_t)\right] = \Phi_t^T \partial_{A_t} \mathbb{E}\left[L(\Phi_t, w_t)\right] = 0. \tag{15}$$

On the other hand, the semi-gradient update for Φ_t can be written as

$$\dot{\Phi}_t = -\partial_{A_t} \mathbb{E}\left[L(\Phi_t, w_t)\right] (w_t)^T.$$

Thanks to Equation (15), we have

$$\Phi_t^T \dot{\Phi}_t = -\Phi_t^T \partial_{A_t} \mathbb{E}\left[L(\Phi_t, P_t)\right] (P_t)^T = 0.$$

Then, taking time derivative on the covariance matrix

$$\frac{d}{dt}\left(\Phi_t^T \Phi_t\right) = \dot{\Phi}_t^T \Phi_t + \Phi_t^T \dot{\Phi}_t = \left(\Phi_t^T \dot{\Phi}_t\right)^T + \Phi_t^T \dot{\Phi}_t = 0,$$

which implies that the covariance matrix is constant along the ODE dynamics. □

Theorem 8. Assume the outer product of the reward function is an identity matrix $R^\pi (R^\pi)^T = I_{|\mathcal{X}| \times |\mathcal{X}|}$ and assume P^π is symmetric. Then under the two time-scale dynamics in Equation (9), the trace objective is non-increasing $df/dt \geq 0$ over time. If $\frac{d\Phi_t}{dt} \neq 0$, then $df/dt > 0$.

Proof. Under the assumption P^π is symmetric, we can verify the uniform distribution $D^\pi = |\mathcal{X}|^{-1} I$ is the stationary distribution. As before, for notational simplicity we let $A = D^\pi (I - \gamma P^\pi)$. Under the reversibility assumption, we have $A^T = A$. Rewriting the TD fixed point with A, we have

$$w^*_\Phi = \left(\Phi^T A \Phi\right)^{-1} \Phi^T A V^\pi.$$

Plugging w^*_Φ into Equation (9), the aggregate dynamics to Φ_t is

$$\frac{d\Phi_t}{dt} = \eta_\Phi \cdot \left(I - A\Phi_t \left(\Phi_t^T A \Phi_t\right)^{-1} \Phi_t^T\right) A V^\pi (V^\pi)^T A^T \Phi_t \left(\Phi_t^T A^T \Phi_t\right)^{-1}.$$

When $R^\pi (R^\pi)^{-1} = I$, we have

$$V^\pi (V^\pi)^T = (I - \gamma P^\pi)^{-1} \left((I - \gamma P^\pi)^{-1}\right)^T = |\mathcal{X}|^{-2} A^{-1} (A^{-1})^T.$$

Plugging this back into the dynamics for Φ_t, we have

$$\frac{d\Phi_t}{dt} = \eta_\Phi |\mathcal{X}|^{-2} \cdot \left(I - A\Phi_t \left(\Phi_t^T A \Phi_t\right)^{-1} \Phi_t^T\right) AA^{-1}(A^{-1})^T A\Phi_t \left(\Phi_t^T A^T \Phi_t\right)^{-1}.$$

Now, consider the trace objective $f(\Phi_t) = \text{Tr}\left(\Phi_t^T (I - \gamma P^\pi)^{-1} \Phi_t\right) = |\mathcal{X}|^{-1} \text{Tr}\left(\Phi_t^T A^{-1} \Phi_t\right)$. Note that since $(\Phi_t^T A^T \Phi_t)^{-1}$ is symmetric and PD, we can write $(\Phi_t^T A^T \Phi_t)^{-1} = LL^T$. Then we consider its time derivative

$$\begin{aligned}
\frac{df}{dt} &= \text{Tr}\left(\Phi_t^T A^{-1} \frac{d\Phi_t}{dt}\right) \\
&= 2|\mathcal{X}|^{-1} \cdot \text{Tr}\left(\Phi_t^T A^{-1} \frac{d\Phi_t}{dt}\right) \\
&= 2\eta_\Phi |\mathcal{X}|^{-3} \cdot \text{Tr}\left(\Phi_t^T A^{-1} \left(I - A\Phi_t \left(\Phi_t^T A \Phi_t\right)^{-1} \Phi_t^T\right) AA^{-1}(A^{-1})^T A\Phi_t \left(\Phi_t^T A^T \Phi_t\right)^{-1}\right) \\
&= 2\eta_\Phi |\mathcal{X}|^{-3} \cdot \text{Tr}\left(\Phi_t^T (\sqrt{A})^{-1} \left(I - \sqrt{A}\Phi_t \left(\Phi_t^T A \Phi_t\right)^{-1} \Phi_t^T \sqrt{A}\right) (\sqrt{A})^{-1} \Phi_t LL^T\right) \\
&= 2\eta_\Phi |\mathcal{X}|^{-3} \cdot \text{Tr}\left(\Phi_t^T (\sqrt{A})^{-1} \left(I - \sqrt{A}\Phi_t \left(\Phi_t^T A \Phi_t\right)^{-1} \Phi_t^T \sqrt{A}\right) \underbrace{(\sqrt{A})^{-1} \Phi_t L}_{=:v} L^T\right) \\
&= 2\eta_\Phi |\mathcal{X}|^{-3} \cdot \text{Tr}\left(v^T \left(I - \underbrace{\sqrt{A}\Phi_t \left(\Phi_t^T A \Phi_t\right)^{-1} \Phi_t^T \sqrt{A}}_{=:M}\right) v\right)
\end{aligned}$$

Note the matrix M is a projection matrix which satisfies $M^T = M, M^T = M$. This means $\text{Tr}\left(v^T (I - M) v\right) \geq 0$ for any matrix v. With the above definitions, we can rewrite the update

$$\frac{d\Phi_t}{dt} = \sqrt{A}(I - M) v L^T.$$

Now, if $\frac{d\Phi_t}{dt} \neq 0$, this implies $(I - M)v \neq 0$ since both LL^T and A are PD. This further implies $\frac{df}{dt}$ is strictly positive. Meanwhile, if $\frac{d\Phi_t}{dt} = 0$, this means $(I - M)v = 0$, which further implies $\frac{df}{dt} = 0$.

□

Corollary 9. When assumptions in Theorem 8 are satisfied, and further, assume $\Phi_0^T D^\pi (I - \gamma P^\pi) \Phi_0$ is invertible. A representation Φ is a non-trivial critical point of the two time-scale dynamics (Equation (9)) if and only if it spans an invariant subspace of P^π, i.e., $P^\pi \Phi \in \text{span}(\Phi)$.

Proof. Define $A = D^\pi (I - \gamma P^\pi)$. When $\Phi_0^T A \Phi_0$ is assumed invertible, and since A is PD, Φ must be of rank k. By Lemma 7, the rank of Φ is preserved over time and hence $\Phi_t^T A \Phi_t$ is always invertible along the ODE dynamics. This means the conditions for Lemma 4 is are all satisfied and hence,

$$D^\pi R^\pi (D^\pi R^\pi)^T \Phi \in \text{span}(A\Phi).$$

When P^π is symmetric and $R^\pi (R^\pi)^T = I$, we deduce $D^\pi = |\mathcal{X}|^{-1} I$ and the following specialized condition holds,

$$\Phi \in \text{span}(A\Phi).$$

Since Φ is rank k, the above condition implies there exists an invertible matrix $B \in \mathbb{R}^{k \times k}$,

$$\Phi = A\Phi B.$$

This in turn implies $A\Phi = \Phi B^{-1}$ and equivalently, $A\Phi \in \text{span}(\Phi)$. When P^π is symmetric, $D^\pi = |\mathcal{X}|^{-1} I$ and we have from before, $\Phi - \gamma P^\pi \Phi \in \text{span}(\Phi)$. This implies $P^\pi \Phi = \gamma^{-1} \Phi (I - B)$ and hence $P^\pi \Phi \in \text{span}(\Phi)$.

Now, assume $P^\pi \Phi \in \text{span}(\Phi)$, we seek to show that Φ is a critical point of the dynamics. The assumption directly implies that there exists matrix $B \in \mathbb{R}^{k\times k}$ such that $P^\pi \Phi = \Phi B$.. This implies

$$(I - \gamma P^\pi)\Phi = \Phi(I - \gamma B).$$

Now, since $\Phi^T\Phi = \Phi_0^T\Phi_0$ and because Φ_0 is initialized of rank k, Φ is of rank k too. Because the matrix $(I - \gamma P^\pi)$ is full rank, it must be that $(I - \gamma B)$ is of rank k as well, and hence invertible. This gives

$$\Phi = (I - \gamma P^\pi)\Phi(I - \gamma B)^{-1} \in \text{span}(A\Phi).$$

□

Lemma 10. (**Randomly sampled rewards**) When the each entry of R_i^π is sampled i.i.d. from any zero-mean distribution with finite variance $\sigma^2/\sqrt{h}$ for some constant $\sigma > 0$, for $1 \le i \le h$. Then as the number of reward columns $h \to \infty$, we have $R^\pi(R^\pi)^T \to \sigma^2 I_{|\mathcal{X}|\times|\mathcal{X}|}$ almost surely.

Proof. Recall R_n^π to be the n-th column of the reward function matrix R^π. We have

$$R^\pi(R^\pi)^T = \sum_{n=1}^{h} R_n^\pi (R_n^\pi)^T$$

Consider the (i, j)-th component of the matrix

$$\left[R^\pi(R^\pi)^T\right](i,j) = \sum_{n=1}^{h} \left[R_n^\pi(R_n^\pi)^T\right](i,j) = \sum_{n=1}^{h} R_{ni}^\pi R_{nj}^\pi,$$

where R_{ni}^π denotes the i-th component of the column vector R_n^π for $1 \le i \le |\mathcal{X}|$. When $i = j$, the summation is over a set of h i.i.d. random variables each with mean σ^2/h. When $i \ne j$, the summation is over a set of h i.i.d. random variables each with mean zero. The law of large number concludes the proof. □

C. Discussion on related work

Here, we provide a more comprehensive discussion on prior work.

From linear TD to end-to-end linear TD. Since the introduction of TD-learning (Sutton, 1988), there have been numerous efforts at understanding the algorithm. The seminal work of (Tsitsiklis and Van Roy, 1996) proposes to understand linear TD through its expected continuous time behavior, characterized by a linear ODE system. A number of follow-up work has applied similar techniques to understanding the stability of, e.g., Q-learning (Melo et al., 2008), off-policy TD (Sutton et al., 2016), among others problems.

Much of the prior work has focused on the classic linear TD setup, where the representations are assumed fixed throughout learning. Closely related to our work is (Lyle et al., 2021) where they proposed to understand the learning dynamics of end-to-end linear TD through its corresponding ODE system. Since such an ODE system is highly non-linear, it is more challenging to provide generic characterizations without restrictive assumptions. Lyle et al. (2021) bypasses the non-linearity issue by essentially assuming a fixed weight parameter $w_t \equiv w$. In light of Equation (6), they study the following dynamics

$$\frac{d\Phi_t}{dt} = \eta_\Phi \cdot D^\pi \left(R^\pi - (I - \gamma P^\pi)\Phi_t w\right) w^T$$

which effectively reduces to a linear system in Φ_t and is more amenable to analysis.

Our work differs in a few important aspects. Firstly, our analysis adheres strictly to the vanilla end-to-end linear TD (Equation (6)) or two-time scale end-to-end linear TD (Equation (9)), without imposing the constant weight assumption as in (Lyle et al., 2019). Our analysis is also slightly more general, as it consists in constructing a few scalar functions that characterize the dynamics. This is more applicable to general ODEs where obtaining exact solutions is not tractable.

TD with non-linear function approximations. Going beyond the linear parameterization, a number of prior work considered the problem of non-linear TD where the value function is represented as some generic smooth non-linear functions. For example, Maei et al. (2009) established the first convergence result of smooth function classes with two time-scale updates. A number of concurrent work Agazzi and Lu (2022); Cai et al. (2019); Sirignano and Spiliopoulos (2022), each with subtly different theoretical setups in place, showed convergence of TD-learning in the over-parameterized regimes, e.g., when the value functions are approximated by very wide neural networks. In this case, representations are confined to be near the initialized values and cannot evolve much over time. To address the limitation, Zhang et al. (2020) generalized the result from (Cai et al., 2019) and showed that in the over-parameterized regimes, the representations converge to the near optimal ones with the lowest projected value prediction error. Note that Theorem 3 echoes this result and provides a complementary result under the end-to-end linear TD setup.

Another closely related work is (Brandfonbrener and Bruna, 2019), where they established the convergence behavior of TD-learning with smooth homogeneous functions. A key requirement underlying their result is that the MDP is sufficiently reversible, which echos the assumption we make in Theorem 3. However, under their framework there is no clear notion of representation as defined in the bi-linear case. An interesting future direction would be to study representation dynamics with non-linear function approximations.

Representations learning via self-predictive learning. Motivated by value-based TD-learning, self-predictive learning directly employs the notion of bootstrapping to the representation space, which has produced a number of empirically successful implementations (Guo et al., 2020; Schwarzer et al., 2021; Guo et al., 2020). The high level idea is to minimize the prediction error

$$\left\|P\left(\phi_{x_t}\right)-\operatorname{sg}\left(\phi_{x_{t+1}}\right)\right\|_2^2,$$

where $P:\mathbb{R}^k\to\mathbb{R}^k$ is the learned transition dynamics in the representation space. Recently, Tang et al. (Tang et al., 2022) proposes to understand the behavior of self-predictive learning through its corresponding ODE. From their discussion, we note that two time-scale dynamics is a generic way to enforcing optimization constraints. Our results in Section 5 build on this and show a similar characterization for the end-to-end linear TD dynamics.

Two time-scale learning dynamics. The idea of two time-scale learning dynamics is not new in TD-learning. Levine et al. (2017) proposed a combination of LSTD updates for the linear weights on top of DQN representations, which are shaped by gradient descents. In continuous time, this is effectively the two time-scale dynamics (Eqn (9)). While their discussion is empirically motivated, we focus on the representation learning aspect of such a learning dynamics.

Auxiliary tasks. In deep RL literature, it has been empirically observed that certain extra training objectives, normally referred to as *auxiliary tasks*, are useful for improving the agent performance on the objective of interest such as optimizing the cumulative returns (Jaderberg et al., 2016; Bellemare et al., 2019; Dabney et al., 2021). Though the conventional wisdom is that such auxiliary tasks are useful in shaping representations, it is generally difficult to characterize exactly what representations are induced, despite some recent theoretical efforts (Lyle et al., 2021). Our results demonstrate the potential benefits of random reward predictions when combined with TD-learning (Dabney et al., 2021), by connecting the learning dynamics to gradient-based spectral decomposition of the transition matrix.

D. Experiment details

We provide further experimen details on the tabular and deep RL experiments in the main paper.

D.1. Tabular experiments

All tabular experiments are conducted on random MDPs with $|\mathcal{X}|=30$ states. The general transition matrix P^π is generated as follows,

$$P^\pi=\alpha P_{\text{perm}}+(1-\alpha)P_{\text{ds}},$$

where P_{ds} is a randomly sampled doubly-stochastic matrix and P_{perm} is a randomly generated permutation matrix. We set $\alpha=0.95$ so that P^π is likely to violate the reversibility assumption. In the symmetric case, the transition matrix is

computed as

$$P^\pi = (P_{\text{ds}} + P_{\text{ds}}^T)/2.$$

The doubly-stochastic matrix is randomly generated based on the procedure in (Tang et al., 2022). Each entry of the reward function R^π is randomly sampled from $\mathcal{N}(0, 1)$

Normalized trace objective. For any matrix P^π, the trace objective is computed as $f(\Phi_t) = \text{tr}\left(\Phi_t^T A \Phi_t\right)$ for $A = (I - \gamma P^\pi)^{-1}$. To calculate the normalized objective, we compute the baseline value $\bar{f}$ as the sum of the top k eigenvalues of the symmetrized matrix $(A + A^T)/2$. The normalized trace objective is $\bar{f}(\Phi_t) \coloneqq f(\Phi_t)/\bar{f}$. When P^π is symmetric, $\bar{f}(\Phi_t)$ is upper bounded by 1.

Unrolling the ODE dynamics. All results are based on the solving the exact ODE dynamics, using the Scipy ODE solver (Virtanen et al., 2020). Throughout, we initialzie the representation Φ_t matrix as orthonormal. We start by generating k column vectors of size $\mathbb{R}^{|\mathcal{X}|}$, with each entry randomly generated from $\mathcal{N}(0, 1)$. Then we apply the Gram-Schmidt orthogonalization procedure to the columns to compute the initialized representation.

D.2. Deep RL experiments

We use DQN (Mnih et al., 2013) as a baseline and evaluate all algorithmic variants over 15 games in the Atari game suite (Bellemare et al., 2013). Our testbed is a subset of 15 Atari games (Bellemare et al., 2013) on which it has been shown that DQN can achieve reasonable performance, see e.g., (Schaul et al., 2015) for how they select the subset of the games: `asterix`, `boxing`, `breakout`, `freeway`, `gopher`, `gravitar`, `hero`, `ms pacman`, `pong`, `qbert`, `riverraid`, `seaquest`, `skiing`, `space invaders` and `venture`.

Random reward functions. To generate random reward functions $R_i^\pi(x, a), 1 \leq i \leq h$, we initialize h networks randomly and directly use the outputs This is similar to the practice of (Dabney et al., 2021) except that we do not apply additional activations (such as sigmoid or tanh as done in (Dabney et al., 2021)) on top. Throughout, we use the Adam optimizer (Kingma and Ba, 2015) with a fixed learning rate η, see (Mnih et al., 2013) for details of other hyper-parameters.

Hyper-parameters. We tune the learning rate $\eta \in \{0.00025, 0.0001, 0.00005\}$ as suggested in (Dabney et al., 2021). The default DQN uses $\eta = 0.00025$. We find that at $\eta = 0.0001$ the tuned DQN performs the best. For the auxiliary task, we tune the number of random rewards $h \in \{4, 16, 64, 256\}$. We find that $h = 16$ performs slightly better than other alternatives.

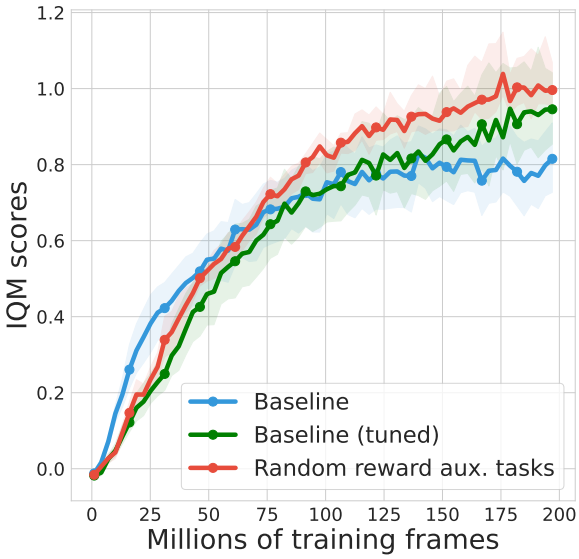

Figure 5. Comparison of training curves of IQM human normalized scores across 15 Atari games. We compare the baseline DQN, DQN with tuned learning rate and DQN with random value predictions as auxiliary tasks. The shaded areas show the 95% bootstrapped confidence intervals averaged over 3 seeds. The random value prediction tasks seem to provide marginal benefits over the tuned DQN. The IQM score is computed by truncating the top and bottom 25% of scores, averaged across all games and all seeds (Agarwal et al., 2021).