Composing Value Functions in Reinforcement Learning

Benjamin van Niekerk [*1] **Steven James** [*1] **Adam Earle** [1] **Benjamin Rosman** [1 2]

Abstract

An important property for lifelong-learning agents is the ability to combine existing skills to solve new unseen tasks. In general, however, it is unclear how to compose existing skills in a principled manner. Under the assumption of deterministic dynamics, we prove that optimal value function composition can be achieved in entropy-regularised reinforcement learning (RL), and extend this result to the standard RL setting. Composition is demonstrated in a high-dimensional video game, where an agent with an existing library of skills is immediately able to solve new tasks without the need for further learning.

1. Introduction

A major challenge in artificial intelligence is creating agents capable of leveraging existing knowledge for inductive transfer (Taylor & Stone, 2009). Lifelong learning, in particular, requires that an agent be able to act effectively when presented with new, unseen tasks.

One promising approach is to combine behaviours learned in various separate tasks to create new skills. This compositional approach allows us to build rich behaviours from relatively simple ones, resulting in a combinatorial explosion in the agent's abilities (Saxe et al., 2017).

Additionally, composition allows us to incrementally expand an agent's abilities—an important property for lifelong learning. Consider, for example, a robot in a warehouse that has the ability to pack objects on shelves. Given a new object, we would want to avoid retraining the robot from scratch. Rather, we would like the robot to learn the skill of packing only the new object, and then compose it with its previous abilities.

In reinforcement learning (RL), one existing approach to composition are linearly-solvable Markov Decision Processes (LMDPs) (Todorov, 2007), which structure the reward function to ensure that the Bellman equation becomes linear in the exponentiated value function. Todorov (2009) proves that the optimal value functions of a set of LMDPs can be composed to produce the optimal value function for a composite task. This is a particularly attractive property, since solving new tasks requires no further learning. However, the LMDP framework has so far been restricted to the tabular case with known dynamics, limiting its usefulness.

Related work has focused on entropy-regularised RL (Haarnoja et al., 2017; Schulman et al., 2017; Nachum et al., 2017), where rewards are augmented with an entropy-based penalty term. This has been shown to lead to improved exploration and rich, multimodal value functions. Haarnoja et al. (2018) demonstrate that these value functions can be composed to *approximately* solve the intersection of tasks. We complement these results by proving *optimal* composition for the union of tasks in the total-reward, absorbing-state setting. Thus, any task that can be expressed as a combination of a set of base tasks can be solved immediately, without any further learning. We provide a formula for optimally composing value functions, and demonstrate our method in a video game. Results show that an agent is able to compose existing policies learned from high-dimensional pixel input to generate new, optimal behaviours.

2. Background

A Markov decision process (MDP) is defined by the 4-tuple $(\mathcal{S}, \mathcal{A}, \rho, r)$ where (i) the *state space* $\mathcal{S}$ is standard Borel; (ii) the *action space* $\mathcal{A}$ is finite (and therefore a compact metric space when equipped with the discrete metric); (iii) the transition dynamics ρ define a Markov kernel $(s, a) \mapsto \rho_{(s,a)}$ from $\mathcal{S} \times \mathcal{A}$ to $\mathcal{S}$; and (iv) the reward r is a real-valued function on $\mathcal{S} \times \mathcal{A}$ that is bounded and measurable.

In RL, an agent's goal is to maximise its utility by making a sequence of decisions. At each time step, the agent receives an observation from $\mathcal{S}$ and executes an action from $\mathcal{A}$ according to its *policy*. As a consequence of its action, the agent receives feedback (reward) and transitions to a new

[*]Equal contribution [1]School of Computer Science and Applied Mathematics, University of the Witwatersrand, Johannesburg, South Africa [2]Council for Scientific and Industrial Research, Pretoria, South Africa. Correspondence to: Benjamin van Niekerk <benjamin.vanniekerk@students.wits.ac.za>, Steven James <steven.james@wits.ac.za>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

state. Whereas the rewards represent only the immediate outcome, the utility captures the long-term consequences of actions. Historically, many utility functions have been investigated (Puterman, 2014), but in this paper we consider the total-reward criterion (see Section 2.1).

We consider the class of MDPs with an absorbing set $\mathcal{G}$, which is a Borel subset of the state space. We augment the state space with a virtual state g such that $\rho_{(s,a)}(\{g\}) = 1$ for all (s, a) in $\mathcal{G} \times \mathcal{A}$, and $r = 0$ after reaching g. In the control literature, this class of MDPs is often called stochastic shortest path problems (Bertsekas & Tsitsiklis, 1991), and naturally model domains that terminate after the agent achieves some goal.

We restrict our attention to stationary Markov policies, or simply policies. A policy $s \mapsto \pi_s$ is a Markov kernel from $\mathcal{S}$ to $\mathcal{A}$. Together with an initial distribution ν over $\mathcal{S}$, a policy defines a probability measure over trajectories. To formalise this, we construct the set of n-step histories inductively by defining $\mathcal{H}_0 = \mathcal{S}$ and $\mathcal{H}_n = \mathcal{H}_{n-1} \times \mathcal{A} \times \mathcal{S}$ for n in $\mathbb{N}$. The n-step histories represent the set of all possible trajectories of length n in the MDP. The probability measure on $\mathcal{H}_n$ induced by the policy π is then

$$P^\pi_{\nu,n} = \nu \otimes \underbrace{\pi \otimes \rho \otimes \cdots \otimes \pi \otimes \rho}_{n \text{ times}}.$$

Using the standard construction (Klenke, 1995), we can define a unique probability measure P^π_ν on $\mathcal{H}_\infty$ consistent with the measures $P^\pi_{\nu,n}$ in the sense that

$$P^\pi_\nu(\mathcal{E} \times \mathcal{A} \times \mathcal{S} \times \mathcal{A} \times \cdots) = P^\pi_{\nu,n}(\mathcal{E}),$$

for any n in $\mathbb{N}$ and any Borel set $\mathcal{E} \subseteq \mathcal{H}_n$. If ν is concentrated on a single state s, we simply write $P^\pi_\nu = P^\pi_s$. Additionally for any real-valued bounded measurable function f on $\mathcal{H}_n$, we define $\mathbb{E}^\pi_\nu[f]$ to be the expected value of f under P^π_ν.

Finally, we introduce the notion of a *proper policy*—a policy under which the probability of reaching $\mathcal{G}$ after n steps converges to 1 uniformly over $\mathcal{S}$ as $n \to \infty$. Our definition extends that of Bertsekas & Tsitsiklis (1995) to general state spaces, and is equivalent to the definition of transient policies used by James & Collins (2006):

Definition 1. A stationary Markov policy π is said to be *proper* if

$$\sup_{s \in \mathcal{S}} \sum_{t=0}^{\infty} P^\pi_s(s_t \notin \mathcal{G}) < \infty.$$

Otherwise, we say that π is improper.

2.1. Entropy-Regularised RL

In the standard RL setting, the expected reward at state s under policy π is given by $\mathbb{E}_{a\sim\pi}[r(s, a)]$. Entropy-regularised RL (Ziebart, 2010; Fox et al., 2016; Haarnoja et al., 2017; Schulman et al., 2017; Nachum et al., 2017) augments the reward function with a term that penalises deviating from some reference policy $\bar{\pi}$. That is, the expected reward is given by $\mathbb{E}_{a\sim\pi}[r(s, a)] - \tau \mathrm{KL}[\pi_s || \bar{\pi}_s]$, where τ is a positive scalar temperature parameter and $\mathrm{KL}[\pi_s||\bar{\pi}_s]$ is the Kullback-Leibler divergence between π and the reference policy $\bar{\pi}$ at state s. When $\bar{\pi}$ is the uniform random policy, the regularised reward is equivalent to the standard entropy bonus up to an additive constant (Schulman et al., 2017). This results in policies that are able to preserve multimodality when faced with a task that can be solved in multiple different ways (Haarnoja et al., 2017). Additionally, the reference policy can be used to encode prior knowledge through expert demonstration.

Based on the above regularisation, we define the n-step value function starting from s and following policy π as:

$$V_{\pi,n}(s) = \mathbb{E}^\pi_s \left[\sum_{t=0}^{n-1} r(s_t, a_t) - \tau \mathrm{KL}[\pi_{s_t} || \bar{\pi}_{s_t}] \right].$$

Note that since the KL-divergence term is measurable (Dupuis & Ellis, 2011, Lemma 1.4.3), $V_{\pi,n}$ is well-defined. The infinite-horizon value function, which represents the total expected return after executing π from s, is then

$$V_\pi(s) = \limsup_{n\to\infty} V_{\pi,n}(s).$$

Since the reward function and KL-divergence are bounded,[1] V_π is well defined. Similarly, we define the Q-function to be the expected reward after taking action a in state s, and thereafter following policy π:

$$Q_\pi(s, a) = r(s, a) + \int_{\mathcal{S}} V_\pi(s')\rho_{(s,a)}(ds'). \quad (1)$$

Given the definitions above, we say that a measurable function V^* is optimal if $V^*(s) = \sup_\pi V_\pi(s)$ for all s in $\mathcal{S}$. Furthermore, a policy π^* is optimal if $V_{\pi^*} = V^*$.

In the standard RL case, the optimal policy is always deterministic and is defined by $\mathrm{argmax}_a Q^*(s, a)$. On the other hand, entropy-regularised problems may not admit an optimal deterministic policy. This owes to the KL-divergence term, which penalises deviation from the reference policy $\bar{\pi}$. If $\bar{\pi}$ is stochastic, then a deterministic policy may incur more cost than a stochastic policy.

3. Soft Value and Policy Iteration

The composition results to be discussed in Section 4 hold only in total-reward, entropy-regularised MDPs defined in

[1] Under the assumptions that $\mathcal{A}$ is finite and $\bar{\pi}$ is chosen so that π_s is absolutely continuous with respect to $\bar{\pi}_s$ for any state s and policy π.

Section 2. While value and policy iteration in entropy-regularised RL have been analysed previously (Nachum et al., 2017), convergence results are limited to discounted MDPs. Therefore, in this section, we sketch an argument that an optimal proper policy exists under the total-reward criterion and that the entropy-regularised versions of value and policy iteration (see Algorithms 1 and 2) converge to optimal solutions. More details can be found in the supplementary material.

We begin by defining the Bellman operators:

$$[\mathcal{T}_\pi V_\pi](s) = \int_{\mathcal{A}} Q_\pi(s,a)\pi_s(da) - \tau \mathrm{KL}[\pi_s||\bar{\pi}_s], \quad (2)$$

$$[\mathcal{T}V](s) = \sup_\pi [\mathcal{T}_\pi V](s). \quad (3)$$

Equations (2) and (3) are analogous to the standard Bellman operator and Bellman optimality operator respectively. Note that since the optimal policy may not be deterministic, the Bellman optimality operator selects the supremum over policies instead of actions.

We also define the soft Bellman operator

$$[\mathcal{L}V_\pi](s) = \tau \log \int_{\mathcal{A}} \exp\left(Q_\pi(s,a)/\tau\right) \bar{\pi}_s(da). \quad (4)$$

Here $\mathcal{L}$ is referred to as "soft", since it is a smooth approximation of the $\max$ operator. The soft Bellman operator is connected to the Bellman optimality operator through the following result:

Lemma 1. *Let $V : \mathcal{S} \to \mathbb{R}$ be a bounded measurable function. Then $\mathcal{T}V = \mathcal{L}V$ and the supremum is attained uniquely by the Boltzmann policy $\mathcal{B}[V]$ defined by*

$$\frac{d\mathcal{B}_s[V]}{d\bar{\pi}_s}(a) = \frac{\exp\left(Q(s,a)/\tau\right)}{\int_A \exp\left(Q(s,a')/\tau\right)\bar{\pi}(da'|s)}.$$

Proof. Follows directly from Dupuis & Ellis (2011, Proposition 1.4.2). □

Analogous to the standard RL setting, we can define value and policy iteration in the entropy-regularised context, where the Bellman operators are replaced with their "soft" equivalents:

Algorithm 1 Soft Value Iteration

Input: MDP, temperature $\tau > 0$, bounded function V
Output: Optimal value function V^*
initialize $V^* \leftarrow V$
repeat
  replace $V \leftarrow V^*$
  apply soft Bellman operator $V^* \leftarrow \mathcal{L}[V]$
until convergence

Algorithm 2 Soft Policy Iteration

Input: MDP, temperature $\tau > 0$, proper policy π
Output: Optimal policy π^*
initialize $\pi^* \leftarrow \pi$
repeat
  replace $\pi \leftarrow \pi^*$
  policy evaluation:
    find V_π, the fixed-point of $\mathcal{T}_\pi$
  policy improvement:
    compute the Boltzmann policy $\pi^* \leftarrow \mathcal{B}[V_\pi]$
until convergence

In order to prove the convergence of the above algorithms in the total-reward setting, we require the following assumptions:

Assumption 1. *Suppose that the following hold:*

(i) the map $a \mapsto r(s,a)$ is upper semicontinuous for all s in $\mathcal{S}$; and
(ii) for every state s in $\mathcal{S}$ and every bounded measurable function f, the map

$$a \mapsto \int_{\mathcal{S}} f(s')\rho_{(s,a)}(ds')$$

is continuous.

These conditions on the transition dynamics and reward function are trivially satisfied if $\mathcal{S}$ is countable. We require a further two assumptions:

Assumption 2. *Suppose that the following hold:*

(i) there exists at least one proper continuous policy; and
(ii) if a policy is improper, then its value function is unbounded below.

Along with the compactness of $\mathcal{A}$, these assumptions are identical to those of James & Collins (2006).

We now proceed with the main result of this section. The proof follows closely along the lines of Bertsekas & Tsitsiklis (1991) and James & Collins (2006), but special care is taken to account for the fact that optimal policies are not necessarily deterministic.

Theorem 1. *Suppose that Assumptions 1 and 2 hold and that the optimal value function is bounded above. Then:*

(i) there exists an optimal proper policy;
(ii) the optimal value function is the unique bounded measurable solution to the optimality equation;
(iii) the soft policy iteration algorithm converges to the policy starting from any proper policy;
(iv) the soft value iteration algorithm converges to the optimal value function starting from any proper policy.

Proof. See supplementary material. □

4. Compositionality

In a lifelong learning context, an agent is presented with a series of tasks drawn from some distribution. The agent's goal is to exploit knowledge gained in previous tasks to improve performance in the current task. We consider an environment with fixed state space $\mathcal{S}$, action space $\mathcal{A}$, deterministic transition dynamics ρ, and absorbing set $\mathcal{G}$. Let $\mathcal{D}$ be a fixed but unknown distribution over $(\mathcal{S}, \mathcal{A}, \rho, r)$. The agent is then presented with tasks sampled from $\mathcal{D}$, which differ only in their reward functions—that is, a task is directly specified by its reward function. In this section, we describe a compositional approach to tackle this problem.

4.1. –OR– Composition

Suppose further that the reward functions drawn from $\mathcal{D}$ differ only on the absorbing set $\mathcal{G}$. This restriction was introduced by Todorov (2009), and is a strict subset of the *successor representations* framework (Dayan, 1993; Barreto et al., 2017).

The composition we describe in this section can be viewed as –OR– task composition: if objectives of two tasks are to achieve goals A and B respectively, then we aim to compose the individual Q-functions to solve A–OR–B. To construct the composed –OR– task, we take the maximum (or soft-maximum) of the reward functions of the composite tasks. We now show that, having encountered a set of tasks, we can combine the library of previously-learned Q-functions to solve any new tasks that lies in their "span", without the need for further learning:

Theorem 2 (Optimal Composition)**.** *Let $\mathcal{M}_1, \ldots, \mathcal{M}_m$ be a library of tasks drawn from $\mathcal{D}$. Let $Q^{*,k}_\tau$ be the optimal entropy-regularised Q-function, and r_k be the reward function for $\mathcal{M}_k$. Define the functions $\mathbf{r}$ and $\mathbf{Q}^*_\tau : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^m$ by:*

$$\mathbf{r} = [r_1, \ldots, r_m] \quad \text{and} \quad \mathbf{Q}^*_\tau = [Q^{*,1}_\tau, \ldots, Q^{*,m}_\tau].$$

Given a set of non-negative weights $\mathbf{w}$, with $||\mathbf{w}||_1 = 1$, consider a further task drawn from $\mathcal{D}$ with reward function satisfying

$$r(s,a) = \tau \log \left(|| \exp(\mathbf{r}(s,a)/\tau) ||_{\mathbf{w}} \right) \tag{5}$$

for all s in $\mathcal{G}$, where $||\cdot||_{\mathbf{w}}$ denotes the weighted 1-norm. Then the optimal Q-value for this task is given by:

$$Q^*_\tau(s,a) = \tau \log \left(|| \exp(\mathbf{Q}^*_\tau(s,a)/\tau) ||_{\mathbf{w}} \right). \tag{6}$$

That is, the optimal Q-functions for the library of tasks can be composed to form Q^*_τ.

Proof. Since ρ is deterministic, we can find a measurable function $f : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ such that $\rho_{(s,a)} = \delta_{f(s,a)}$. For any Q-function, define the *desirability function*

$$Z(s,a) = \exp\left(Q(s,a)/\tau\right),$$

and define the operator $\mathcal{U}$ on the space of non-negative bounded measurable functions by

$$[\mathcal{U}Z](s,a) = \exp\left(r(s,a)/\tau\right) \int_{\mathcal{A}} Z(f(s,a),a') \bar{\pi}_s(da').$$

We now show that the desirability function is a fixed point of $\mathcal{U}$.

Since V^*_τ is the fixed point of the Bellman optimality operator, by combining Lemma 1 and Theorem 1 we have

$$V^*_\tau(s) = [\mathcal{T}V^*](s) = [\mathcal{L}V^*](s).$$

Then using the definition of the soft Bellman operator:

$$V^*_\tau(s) = \tau \log \int_{\mathcal{A}} \exp\left(Q^*_\tau(s,a')/\tau\right) \bar{\pi}_s(da').$$

Additionally, under the assumption of a deterministic environment, Equation (1) can be rewritten as

$$Q^*_\tau(s,a) = r(s,a) + V^*_\tau(f(s,a)).$$

Then it follows that

$$\begin{aligned}
[\mathcal{U}Z^*_\tau](s,a) &= \exp\left(r(s,a)/\tau\right) \int_{\mathcal{A}} Z^*_\tau(f(s,a),a') \bar{\pi}_s(da') \\
&= \exp\left(r(s,a)/\tau\right) \int_{\mathcal{A}} \exp\left(Q^*_\tau(f(s,a),a')/\tau\right) \bar{\pi}_s(da') \\
&= \exp\left(r(s,a)/\tau\right) \exp\left(V^*_\tau(f(s,a))/\tau\right) \\
&= \exp\Big(\big(r(s,a) + V^*_\tau(f(s,a))\big)/\tau \Big) \\
&= \exp\left(Q^*_\tau(s,a)/\tau\right) \\
&= Z^*_\tau(s,a).
\end{aligned}$$

Hence Z^*_τ is a fixed point of $\mathcal{U}$.

Given a task $\mathcal{M}_k$ and terminal state s in $\mathcal{G}$, the optimal Q-value at that state is simply $r_k(s,a)$. Therefore, for the combined task with reward function (5), the optimal Q-value satisfies

$$\begin{aligned}
Q^*_\tau(s,a) &= \tau \log \left(|| \exp(\mathbf{r}(s,a)/\tau) ||_{\mathbf{w}} \right) \\
&= \tau \log \left(|| \exp(\mathbf{Q}^*_\tau(s,a)/\tau) ||_{\mathbf{w}} \right)
\end{aligned}$$

on $\mathcal{G}$. Thus, restricted to $\mathcal{G}$, the desirability function Z^*_τ is a linear combination of the desirability functions for the family of tasks.

Finally, since (6) holds on $\mathcal{G}$ and it is clear that $\mathcal{U}$ is a linear operator on the exponentiated Q-function, then (6) holds everywhere. □

The following lemma links the previous result to the standard RL setting. Recall that entropy-regularisation appends a temperature-controlled penalty term to the reward function. As the temperature parameter tends to 0, the reward provided by the environment dominates the entropy penalty, and the problem reduces to the standard RL case:

Lemma 2. *Let $\{\tau_n\}_{n=1}^{\infty}$ be a sequence in $\mathbb{R}$ such that $\tau_n \downarrow 0$. Let $Q^*_{\tau_n}$ be the optimal Q-value function for MDP(τ_n): the entropy-regularised MDP with temperature parameter τ_n. Let Q^*_0 be the optimal Q-value for the standard MDP. Then $Q^*_{\tau_n} \uparrow Q^*_0$ as $n \to \infty$.*

Proof. First note that for a fixed policy π, state s and action a, we have $Q^\pi_{\tau_n}(s,a) \uparrow Q^\pi_0(s,a)$ as $n \to \infty$. This follows directly from the definition of the entropy-regularised value function, and the fact that the KL-divergence is non-negative. Then using Lemma 3.14 (Hinderer, 1970) to interchange the limit and supremum, we have

$$\begin{aligned}\lim_{n\to\infty} Q^*_{\tau_n} &= \lim_{n\to\infty} \sup_\pi Q^\pi_{\tau_n} = \sup_\pi \lim_{n\to\infty} Q^\pi_{\tau_n} \\ &= \sup_\pi Q^\pi_0 = Q^*_0.\end{aligned}$$

Since $Q^\pi_{\tau_n} \uparrow Q^\pi_0$, we have $Q^*_{\tau_n} \uparrow Q^*_0$ as $n \to \infty$. □

We can now show that composition holds in the standard RL setting by using Lemma 2 to take the low-temperature limit of Theorem 2.

Corollary 1. *Let $\{\tau_n\}_{n=1}^{\infty}$ be a sequence in $\mathbb{R}$ such that $\tau_n \downarrow 0$ and let Q^*_0 be the optimal Q-function for the standard MDP with a composite reward satisfying $r = \max \mathbf{r}$. Then $\max \mathbf{Q}^*_{\tau_\mathbf{n}} \uparrow Q^*_0$ as $n \to \infty$.*

Proof. For a fixed state s and action a and a possible reordering of the vector $\mathbf{Q}^*_\mathbf{0}(s,a)$, we may suppose, without loss of generality, that $Q^{*,1}_0(s,a) = \max \mathbf{Q}^*_\mathbf{0}(s,a)$. Then by Lemma 2, we can find an N in $\mathbb{N}$ such that

$$Q^{*,1}_{\tau_n}(s,a) = \max \mathbf{Q}^*_{\tau_\mathbf{n}}(s,a) \text{ for all } n \geq N.$$

Since $\log$ is continuous, we have from Theorem 2 that

$$\lim_{n\to\infty} Q^*_{\tau_n} = \log\left(\lim_{n\to\infty} ||\exp(\mathbf{Q}^*_{\tau_\mathbf{n}})||^{1/\tau_n}_\mathbf{w}\right),$$

where $||\cdot||^p_\mathbf{w}$ denotes the weighted p-norm. By factoring $\exp(Q^{*,1}_{\tau_n})$ out of $||\exp(\mathbf{Q}^*_{\tau_\mathbf{n}})||^{1/\tau_n}_\mathbf{w}$, we are left with

$$||1, \exp(\Delta_2), \ldots, \exp(\Delta_k)||^{1/\tau_n}_\mathbf{w},$$

where $\Delta_i = Q^{*,i}_{\tau_n} - Q^{*,1}_{\tau_n}$ for $i = 2, \ldots, k$. Since $Q^{*,1}_{\tau_n}(s,a)$ is the maximum of $\mathbf{Q}^*_{\tau_\mathbf{n}}(s,a)$ for all $n \geq N$, the limit as $n \to \infty$ of the above is 1. Then it follows that

$$\begin{aligned}\lim_{n\to\infty} Q^{*,1}_{\tau_n}(s,a) &= \log\left(\lim_{n\to\infty} \exp(Q^{*,1}_{\tau_n}(s,a))\right) \\ &= Q^{*,1}_0(s,a).\end{aligned}$$

Since s and a were arbitrary and $Q^{*,m}_{\tau_n} \uparrow Q^{*,m}_0$, we have that $\max \mathbf{Q}^*_{\tau_\mathbf{n}} \uparrow Q^*_0$ as $n \to \infty$. □

Comparing Theorem 2 to Corollary 1, we see that as the temperature parameter decreases to zero, the weight vector has less influence on the composed Q-function. In the limit, the optimal Q-function is independent of the weights and is simply the maximum of the library functions. This suggests a natural trade-off between our ability to interpolate between Q-functions, and the stochasticity of the optimal policy.

Finally, we note that the assumption of deterministic dynamics is necessary. To see this, consider an MDP with a fixed start state and three goal states `Red`, `Purple` and `Blue`. The MDP has three actions `a`, `b`, and `c` with transition dynamics given by:

	Red	Purple	Blue
a	0.1	0.8	0.1
b	0.1	0.1	0.8
c	0.0	0.5	0.5

Suppose that tasks A and B are to reach the `Purple` and `Blue` states respectively (with a reward of 1 for getting to the correct state and 0 otherwise). The optimal policy for the composed task A–OR–B is therefore to select action `c`, yielding a return of 1. However, the policy given by Corollary 1 selects either `a` or `b` with a return of 0.9.

4.2. –AND– Composition

Haarnoja et al. (2018) show that an approximate –AND– composition is also possible for entropy-regularised RL. That is, if the goals A and B partially overlap, the composed Q-function will achieve A–AND–B approximately. The following result, included for completeness, demonstrates that the optimal Q-function for the composite task can be approximated by the average of the library Q-functions:

Lemma 3 (Haarnoja et al. (2018))**.** *Let $Q^{*,1}_\tau$ and $Q^{*,2}_\tau$ be the optimal Q-functions for two tasks drawn from $\mathcal{D}$ with rewards r_1 and r_2. Define the averaged Q-function $Q_{ave} := (Q^{*,1}_\tau + Q^{*,2}_\tau)/2$. Then the optimal Q-function Q^*_τ for the task with reward function $r = (r_1 + r_2)/2$ satisfies*

$$Q_{ave} \geq Q^*_\tau \geq Q_{ave} - C^*_\tau,$$

where C^*_τ is a fixed point of

$$\tau \mathbb{E}_{s' \sim \rho(s,a)}\left[D_{\frac{1}{2}}\left(\pi^{*,1}_s || \pi^{*,2}_s\right) + \max_{a'} C(s', a')\right],$$

the policy $\pi^{*,i}_s$ is the optimal Boltzmann policy for task i, and $D_{\frac{1}{2}}(\cdot||\cdot)$ is the Rényi divergence of order $\frac{1}{2}$.

Theorem 3 (Haarnoja et al. (2018))**.** *Using the definitions in Lemma 3, the value of the composed policy π^{ave} satisfies*

$$Q_{\pi^{ave}} \geq Q^*_\tau - F^*_\tau,$$

where F^*_τ is a fixed point of

$$\tau \mathbb{E}_{s' \sim \rho(s,a)} \left[\mathbb{E}_{a' \sim \pi^{ave}_{s'}} \left[C^*_\tau(s', a') - F(s', a') \right] \right].$$

We believe that the low-temperature result from Lemma 2 can be used to obtain similar results for the standard RL framework. We provide empirical evidence of this in the next section, and leave a formal proof to future work.

5. Experiments

To demonstrate composition, we perform a series of experiments in a high-dimensional video game (Figure 1b). The goal of the game is to collect items of different colours and shapes. The agent has four actions that move it a single step in any of the cardinal directions, unless it collides with a wall. Each object in the domain is one of two shapes (squares and circles), and one of three colours (blue, beige and purple), for a total of six objects (see Figure 1a).

We construct a number of different tasks based on the objects that the agent must collect, the task's name specifying the objects to be collected. For example, `Purple` refers to the task where an agent must collect any purple object, while `BeigeSquare` requires collecting the single beige square.

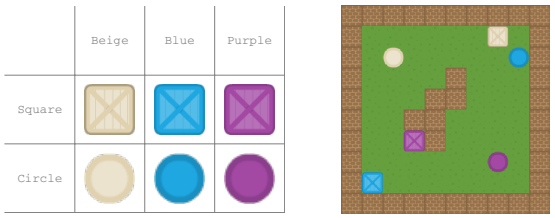
For each task, the episode begins by randomly positioning the six objects and the agent. At each timestep, the agent receives a reward of -0.1. If the correct object is collected, the agent receives a reward of 1 and the episode terminates. We first learn to solve a number of base tasks using (soft) deep Q-learning (Mnih et al., 2015; Schulman et al., 2017), where each task is trained with a separate network. Each network is trained for 1.5m timesteps to ensure near-optimal convergence. The resulting networks are collected into a library from which we will later compose new Q-functions.

The input to our network is a single RGB frame of size 84×84, which is passed through three convolutional layers and two fully-connected layers before outputting the predicted Q-values for the given state.[2] Using the results in Section 4, we compose optimal Q-functions from those in the library.

In all our results, we visualise value functions by placing the agent at each cell in the domain and feeding the resulting state into the learned Q-function. We take the maximum output as the value of the state and then interpolate between cells to form a smooth surface.

5.1. –OR– Composition

Here we consider new tasks that can be described as the union of a set of base tasks in the standard RL setting. We train an agent separately on the `Purple` and `Blue` tasks, adding the corresponding Q-functions to our library. We use Corollary 1 to produce the optimal Q-function for the composite `PurpleOrBlue` task, which requires the agent to pick up either blue or purple objects, without any further learning. Results are given in Figure 2.



(a) Items to be collected. (b) Layout of the grid-world.

Figure 1. The video game domain. The position of the walls and obstacles remains fixed, but at the start of each episode, the agent and objects are randomly positioned.

The local maxima over blue and purple objects illustrates the multimodality of the value function (Figure 2a). This is similar to approaches such as soft Q-learning (Haarnoja et al., 2017), which are also able to learn multimodal policies. However, we have anecdotally observed that directly learning a truly multimodal policy for the composite task can be difficult. If the entropy regularisation is too high, the resulting policy is extremely stochastic. Too low, and the policy quickly collapses to a single mode without exploring alternatives. It is instead far easier to learn unimodal value functions for each of the base tasks, and then compose them to produce optimal multimodal value functions.

5.2. Linear Task Combinations

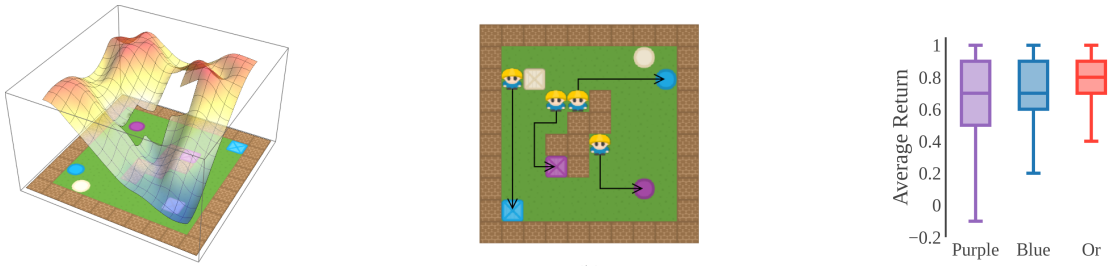
In Theorem 2 we showed that in the entropy-regularised setting, the composed Q-function is dependent on a weight vector $\mathbf{w}$. This allows us to achieve a more general type of composition. In particular, we can immediately compute any optimal Q-function that lies in the "span" of the library Q-functions. Indeed, according to Theorem 2 the exponentiated optimal Q-function is a linear combination of the exponentiated library functions. Therefore, the weights can be used to modulate the relative importance of the library functions—modelling the situation in which an agent has multiple concurrent objectives of unequal importance.

We illustrate the effect of the weight vector $\mathbf{w}$ using soft Q-learning with a temperature parameter $\tau = 1$. We construct a new task by composing the tasks `PurpleCircle` and `BeigeSquare`, and assign different weights to these tasks. The different weighted value functions are given in Figure 3.

5.3. –AND– Composition

Here we consider tasks which can be described as the intersection of tasks in the library. In general, this form of

[2] A full description of the architecture and hyperparameters is provided in the supplementary material.



(a) (b) (c)

Figure 2. (a) The optimal value function for `PurpleOrBlue`, which is produced by composing the `Purple` and `Blue` Q-functions. The multimodality of the composite value function is clearly visible. (b) Sample trajectories for the composite `PurpleOrBlue` task, with the agent beginning at different positions. The agent selects the shortest path to any of the target objects. (c) Returns from 50k episodes. The first two box plots are the results of acting in the `PurpleOrBlue` task using only one of the base Q-functions, while the third uses the composite Q-function.

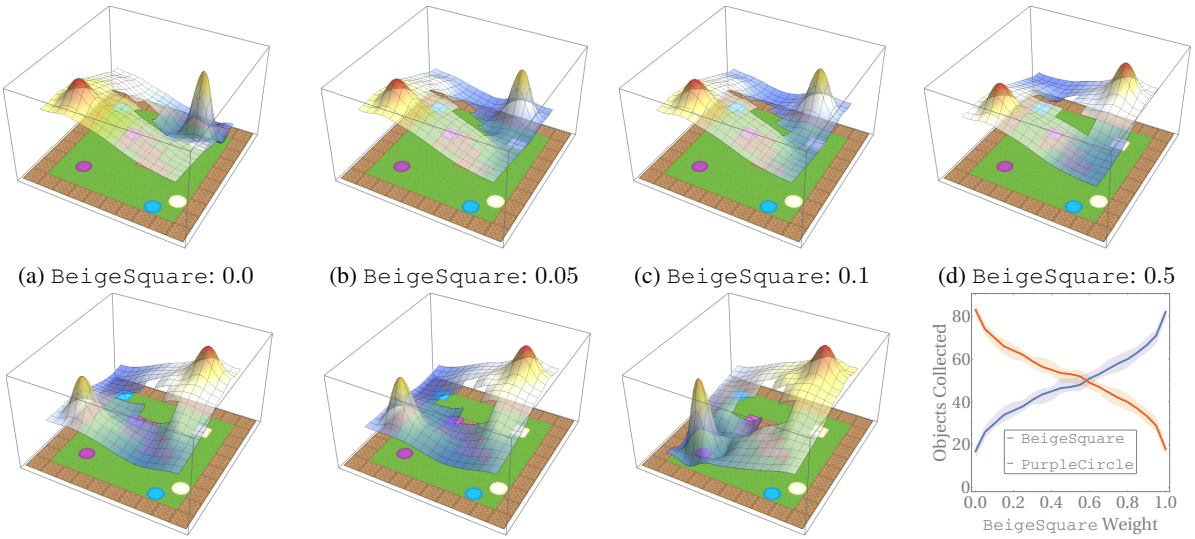
composition will not yield an optimal policy for the composite task owing to the presence of local optima in the composed value function.

However, in many cases we can obtain a good approximation to the composite task by simply averaging the Q-values for the constituent tasks. While Haarnoja et al. (2018) considers this type of composition in the entropy-regularised case, we posit that this can be extended to the standard RL setting by taking the low-temperature limit. We illustrate this by composing the optimal policies for the `Blue` and `Square` tasks, which produces a good approximation to the optimal policy for collecting the blue square. Results are shown in Figure 4.

5.4. Temporal

Our final experiment demonstrates the use of composition to long-lived agents. We compose the base Q-functions for the tasks `Blue`, `Beige` and `Purple`, and use the resulting Q-function to solve the task of collecting *all* objects. Sample trajectories are illustrated by Figure 5.

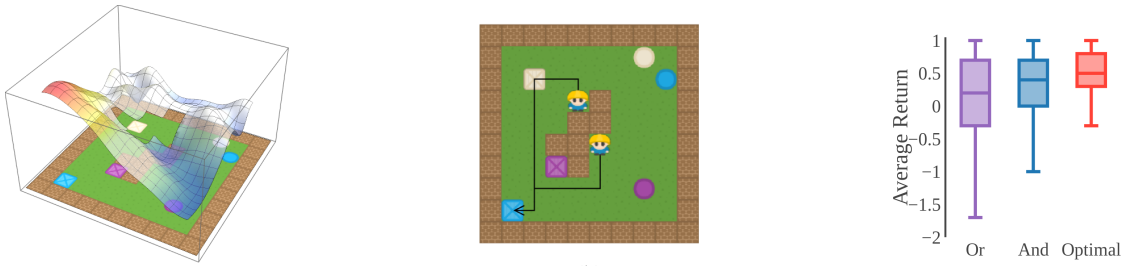
Despite the fact that the individual tasks terminate after collecting the required object, if we allow the episode to continue, the composed Q-function is able to collect all objects in a greedy fashion. The above shows the power of composition—if we possess a library of skills learned from previous tasks, we can compose them to solve any task in their union continually.



(a) `BeigeSquare`: 0.0 (b) `BeigeSquare`: 0.05 (c) `BeigeSquare`: 0.1 (d) `BeigeSquare`: 0.5

(e) `BeigeSquare`: 0.9 (f) `BeigeSquare`: 0.95 (g) `BeigeSquare`: 1.0 (h)

Figure 3. (a–g) Weighted composed value function for the task `BeigeSquareOrPurpleCircle`. The weight assigned to the Q-function for `BeigeSquare` is varied from 0 to 1. (h) The number of beige squares compared to purple circles collected by the agent as the weights are varied in steps of 0.05. Results for each weight were averaged over 80 runs of 100 episodes.



(a) (b) (c)

Figure 4. (a) The approximately optimal value function of the composed policies. Local optima are clearly visible. (b) Sample trajectories from the composed policy beginning from different starting positions. The agent exhibits suboptimal, but sensible behaviour near beige squares. (c) The IQR of returns from 50k episodes. The first box plot is the return from the optimal solution to the *union* of tasks, the second is the result of the approximate *intersection* of tasks, and the third is the true optimal policy.

6. Related Work

As mentioned, the optimal composition described here can be achieved through the LMDP framework (Todorov, 2009). However, LMDPs require the passive dynamics (an $\mathcal{S} \times \mathcal{S}$ matrix) to be specified upfront, which restricts their applicability to low-dimensional settings. Our approach, on the other hand, shows that the same composition can be achieved in both the entropy-regularised and standard RL setting. As a result, we can perform composition in high-dimensional state spaces such as video games.

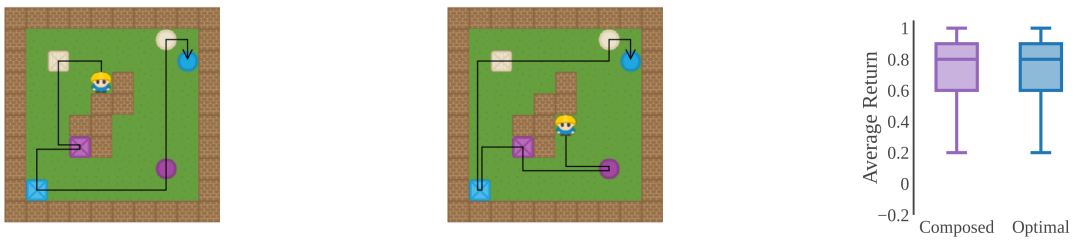
Using the maximimum of a set of previously-learned Q-functions appears in other contexts. Corollary 1 mirrors that of *generalised policy improvement* (Barreto et al., 2017; 2018), which uses the successor representation framework (Dayan, 1993) to show that maximising over Q-functions results in an improved policy. In our work, the resulting Q-function is not merely an improvement, but is in fact optimal. More generally, Abel et al. (2018) provide the *MAXQInit* algorithm, which can be used to solve a series of tasks that differ only in reward function. When faced with a new task, initialising the value function to be the maximum over learned Q-functions is shown to preserve optimism and lower the sample complexity with high probability.

Finally, the composition described here differs from the *options* framework (Sutton et al., 1999), which sequence low-level actions to form high-level skills. Whereas options compose actions temporally, our composition is concurrent, but we note that we are sometimes able to mimic temporal composition (see Figure 5). Since options themselves contain policies, it is likely that they can be composed using the theory described here; however, we would need to account for options whose initiation sets and termination conditions differ.

7. Conclusion

We showed that in entropy-regularised RL, value functions can be optimally composed to solve the union of tasks. Extending this result by taking the low-temperature limit, we showed that composition is also possible in standard RL. However, there is a trade-off between our ability to smoothly interpolate between tasks, and the stochasticity of the optimal policy.

We demonstrated, in a high-dimensional environment, that a library of optimal Q-functions can be composed to solve composite tasks consisting of unions, intersections or temporal sequences of simpler tasks. The proposed compositional framework is a step towards lifelong-learning agents that are able to combine existing skills to solve new, unseen tasks.



(a) (b) (c)

Figure 5. (a) and (b) Sample trajectories for the task of collecting all objects. (c) Returns from 50k episodes. The first box plot is the return of the composed Q-function, while the second is the result of DQN trained to collect all objects explicitly.

Acknowledgements

The authors wish to thank the anonymous reviewers for their thorough feedback and helpful comments. SJ is supported by a Google PhD Fellowship in Machine Learning. BR is supported by a Google Faculty Research Award in Machine Learning.

References

Abel, D., Jinnai, Y., Guo, Y., Konidaris, G., and Littman, M. Policy and value transfer in lifelong reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2018.

Barreto, A., Dabney, W., Munos, R., Hunt, J., Schaul, T., van Hasselt, H., and Silver, D. Successor features for transfer in reinforcement learning. In *Advances in neural information processing systems*, pp. 4055–4065, 2017.

Barreto, A., Borsa, D., Quan, J., Schaul, T., Silver, D., Hessel, M., Mankowitz, D., Žídek, A., and Munos, R. Transfer in deep reinforcement learning using successor features and generalised policy improvement. In *Proceedings of the International Conference on Machine Learning*, pp. 501–510, 2018.

Bertsekas, D. and Tsitsiklis, J. An analysis of stochastic shortest path problems. *Mathematics of Operations Research*, 16(3):580–595, 1991.

Bertsekas, D. and Tsitsiklis, J. Neuro-dynamic programming: an overview. In *Proceedings of the 34th IEEE Conference on Decision and Control*, volume 1, pp. 560–564. IEEE, 1995.

Dayan, P. Improving generalization for temporal difference learning: The successor representation. *Neural Computation*, 5(4):613–624, 1993.

Dupuis, P. and Ellis, R. *A weak convergence approach to the theory of large deviations*. 2011.

Fox, R., Pakman, A., and Tishby, N. Taming the noise in reinforcement learning via soft updates. In *32nd Conference on Uncertainty in Artificial Intelligence*, 2016.

Haarnoja, T., Tang, H., Abbeel, P., and Levine, S. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning*, pp. 1352–1361, 2017.

Haarnoja, T., Pong, V., Zhou, A., Dalal, M., Abbeel, P., and Levine, S. Composable deep reinforcement learning for robotic manipulation. *arXiv preprint arXiv:1803.06773*, 2018.

Hinderer, K. Foundations of non-stationary dynamic programming with discrete time parameter. In *Lecture Notes in Operations Research and Mathematical Systems*, volume 33. 1970.

James, H. and Collins, E. An analysis of transient Markov decision processes. *Journal of applied probability*, 43(3): 603–621, 2006.

Klenke, A. *Probability Theory: A Comprehensive Course*, volume 158. 1995. ISBN 9781447153603.

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

Nachum, O., Norouzi, M., Xu, K., and Schuurmans, D. Bridging the gap between value and policy based reinforcement learning. In *Advances in Neural Information Processing Systems*, pp. 2772–2782, 2017.

Puterman, M. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Saxe, A., Earle, A., and Rosman, B. Hierarchy through composition with multitask LMDPs. *Proceedings of the 34th International Conference on Machine Learning*, 70: 3017–3026, 2017.

Schulman, J., Abbeel, P., and Chen, X. Equivalence between policy gradients and soft Q-learning. pp. 1–15, 2017.

Sutton, R., Precup, D., and Singh, S. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2): 181–211, 1999.

Taylor, M. and Stone, P. Transfer learning for reinforcement learning domains: a survey. *Journal of Machine Learning Research*, 10:1633–1685, 2009.

Todorov, E. Linearly-solvable Markov decision problems. In *Advances in Neural Information Processing Systems*, pp. 1369–1376, 2007.

Todorov, E. Compositionality of optimal control laws. In *Advances in Neural Information Processing Systems*, pp. 1856–1864, 2009.

Ziebart, B. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. Carnegie Mellon University, 2010.