Value Preserving State-Action Abstractions

David Abel
Brown University

Nathan Umbanhowar
Brown University

Khimya Khetarpal
Mila-McGill University

Dilip Arumugam
Stanford University

Doina Precup
Mila-McGill University

Michael L. Littman
Brown University

Abstract

Abstraction can improve the sample efficiency of reinforcement learning. However, the process of abstraction inherently discards information, potentially compromising an agent’s ability to represent high-value policies. To mitigate this, we here introduce combinations of state abstractions and options that are guaranteed to preserve the representation of near-optimal policies. We first define $\phi$-relative options, a general formalism for analyzing the value loss of options paired with a state abstraction, and present necessary and sufficient conditions for $\phi$-relative options to preserve near-optimal behavior in any finite Markov Decision Process. We further show that, under appropriate assumptions, $\phi$-relative options can be composed to induce hierarchical abstractions that are also guaranteed to represent high-value policies.

1 INTRODUCTION

Learning to make high-quality decisions in complex environments is challenging. To mitigate this difficulty, knowledge can be incorporated into learning algorithms through inductive biases, heuristics, or other priors that provide information about the world. In reinforcement learning (RL), one powerful class of such structures takes the form of abstractions, either of state (what are the relevant properties of the world?) or action (what correlated, long-horizon sequences of actions are useful?).

To make RL more tractable, abstractions throw away information about the environment such as irrelevant state features or ineffective sequences of actions. If the abstractions are too aggressive, however, they destroy an agent’s ability to solve tasks of interest. Thus, there is a trade-off inherent in the role that abstractions play: they should 1) make learning easier, while 2) preserving enough information to support the discovery of good behavioral policies. Indeed, prior work has illustrated the potential for abstractions to support both this first (Konidaris and Barto, 2007; Brunskill and Li, 2014; Bacon et al., 2017) and second property (Li et al., 2006; Van Roy, 2006; Hutter, 2014).

To realize the benefits of state and action abstraction, it is common to make use of both. One approach builds around MDP homomorphisms, introduced by Ravindran and Barto (2002), based on the earlier work of Givan et al. (1997) and Whitt (1978). Certain classes of homomorphisms can lead to dramatic reductions in the size of the model needed to describe the environment while preserving representation of high value policies. Ravindran and Barto (2003a, 2004) extend these ideas to semi-Markovian environments based on the options framework (Sutton et al., 1999), illustrating how approximate model reduction techniques can be blended with hierarchical structures to form good compact representation. Majeed and Hutter (2019) carry out similar analysis in non-Markovian environments, again proving which conditions preserve value. However, we lack a general theory of value preserving state-action abstractions, especially when abstract actions are extended over multiple time steps. More concretely, the following question remains open:

Which combinations of state abstractions and options preserve representation of near-optimal policies?

The primary contribution of this work is new analysis clarifying which combinations of state abstractions ($\phi$) and options ($O$) preserve representation of near-optimal policies in finite Markovian environments.
To perform this analysis, we first define $\phi$-relative options, a general formalism for analyzing the value loss of a state abstraction paired with a set of options. We then prove four sufficient conditions, along with one necessary condition, for $\phi$-relative options to preserve near-optimal behavior in any finite Markov Decision Process (MDP) (Bellman, 1957; Puterman, 2014). We further prove that $\phi$-relative options can be composed to induce a hierarchy that preserves near-optimal behavior under appropriate assumptions about the hierarchy’s construction. We suggest these results can support the development of principled methods that learn and make use of value-preserving abstractions.

1.1 Background

We next provide background on state and action abstraction. We take the standard treatment of RL: an agent learns to make decisions that maximize value while interacting with an MDP denoted by the five tuple $(S, A, R, T, \gamma)$. For more on RL, see the book by Sutton and Barto (2018), for more on MDPs, see the book by Puterman (2014).

A state abstraction is an aggregation function that projects the environmental state space into a smaller one. With a smaller state space, algorithms can learn with less computation, space, and samples (Singh et al., 1995; Dearden and Boutilier, 1997; Dietterich, 2000b, a; Andre and Russell, 2002; Jong and Stone, 2005). However, throwing away information about the state space can restrict an agent’s ability to represent good policies; as a trivial illustration of this, consider the maximally-compressing state abstraction that maps all original states in the environment to a single abstract state. Only well-chosen types of state abstraction are known to preserve representation of good policies (Dean and Givan, 1997; Li et al., 2006; Van Roy, 2006; Hutter, 2014; Abel et al., 2016, 2019; Majeed and Hutter, 2019). We define a state abstraction as follows:

Definition 1 (State Abstraction): A state abstraction $\phi : S \rightarrow S_\phi$ maps each ground state, $s \in S$, into an abstract state, $s_\phi \in S_\phi$.

Since the abstract state space is smaller than the original MDP’s state space, we expect to lose information. In the context of sequential decision making, it is natural to permit this information loss so long as RL algorithms can still learn to make useful decisions (Abel et al., 2019; Harutyunyan et al., 2019).

Next, we recall options, a popular formalism for organizing the action space of an agent.

Definition 2 (Option (Sutton et al., 1999)): An option $o \in O$ is a triple $(I_o, \beta_o, \pi_o)$, where $I_o \subseteq S$ is a subset of the state space denoting where the option initiates; $\beta_o \subseteq S$, is a subset of the state space denoting where the option terminates; and $\pi_o : S \rightarrow \Delta(A)$ is a policy prescribed by the option $o$.

Options denote abstract actions; the three components indicate where the option $o$ can be executed ($I_o$), where it terminates ($\beta_o$), and what to do in between these two conditions ($\pi_o$). Options are known to aid in transfer (Konidaris and Barto, 2007, 2009; Brunskill and Li, 2014; Topin et al., 2015), encourage better exploration (Şimşek and Barto, 2004, 2009; Brunskill and Li, 2014; Bacon et al., 2017; Fruit and Lazaric, 2017; Machado et al., 2017; Tiwari and Thomas, 2019; Jinmai et al., 2019), and make planning more efficient (Mann and Mamor, 2014; Mann et al., 2015). We define an action abstraction in terms of options as follows:

Definition 3 (Action Abstraction): An action abstraction is any replacement of the primitive actions, $A$, with a set of options $O$.

Thus, an RL algorithm paired with an action abstraction chooses from among the available options at each time step, runs the option until termination, and then chooses the next option. With $O$ replacing the primitive action space, we are no longer guaranteed to be able to represent every policy and may destroy an agent’s ability to discover a near-optimal policy. If the primitive actions are included, any policy that can be represented in the original state-action space can still be represented. However, by including options and primitive actions, learning algorithms face a larger branching factor, and must search the full space of policies which can hurt learning performance (Jong et al., 2008). Hence, it is often prudent to restrict the action space only to a set of options to avoid blowing up the search space. Ideally, we would make use of options that both 1) preserve representation of good policies, while 2) keeping the branching factor and induced policy class small. Naturally, using the options that always execute the optimal policy maximally satisfies both properties, but are challenging to obtain. Establishing a clear understanding of how this first property can be satisfied under approximate knowledge is the primary motivation of this work.

2 RELATED WORK

The study of abstraction in RL has a long and exciting history, dating back to early work on approximating dynamic programs by Fox (1973) and Whitt (1978, 1979), along with the early work on hierar-
chical RL (Dayan and Hinton, 1993; Wiering and Schmidhuber, 1997; Parr and Russell, 1998; Dietterich, 2000a), options (Sutton et al., 1999), and state abstraction (Tsitsiklis and Van Roy, 1996; Dean and Gil- van, 1997; Andre and Russell, 2002; Li et al., 2006). This literature is vast; we here concentrate on work that is focused on abstractions that aim to retain representation of near-optimal behavior. We divide this summary into each of state, action, joint state-action, and hierarchical abstraction.

**State Abstraction.** The work of Whitt (1978, 1979) paved the way for understanding the value loss of state abstraction in MDPs. Later, Dean and Van Roy (1997) developed a method for finding states that behave similarly to one another via the bisimulation property (Larsen and Skou, 1991). Many subsequent works have explored bisimulation for abstraction, including defining metrics for finite (Ferns et al., 2004; Castro and Precup, 2011; Castro, 2019) and infinite MDPs (Ferns et al., 2006; Taylor et al., 2008). In a similar vein, Li et al. (2006) provide a unifying framework for state abstraction in MDPs, examining when such abstractions will preserve optimal behavior and affect existing convergence guarantees of well-known RL algorithms. Recent work has continued to clarify the conditions under which state abstractions preserve value in MDPs (Jiang et al., 2015a; Abel et al., 2016), non-Markovian environments (Hutter, 2014), planning algorithms (Hostetler et al., 2014; Jiang et al., 2014; Anand et al., 2015) and lifelong RL (Abel et al., 2018). A related and important body of work studies the problem of selecting a state abstraction from a given class (Maillard et al., 2013; Odalric-Ambrym et al., 2013; Jiang et al., 2015a; Ortner et al., 2019).

**Action Abstraction.** Little is known about which kinds of options or action abstractions preserve value; this stems largely from the fact that most prior studies of options consider the setting where options are added to the primitive actions, so $A_{\text{abstract}} = A \cup O$. Since the primitive actions are included, the value loss is always zero as the full space of policies is still representable. Brunskill and Li (2014) study the problem of learning options that improve sample efficiency in lifelong RL. Under mild assumptions, their algorithm finds options that let will match some default sample complexity (without options). However, in order to realize the benefits of options, it is important to consider the removal of primitive actions—otherwise, RL algorithms can still represent the full space of policies, and so learning speed may suffer (Jong et al., 2008).

A few previous works study value preservation in the case where options replace the primitive actions. Most recently, Mann and Mannor (2014); Mann et al. (2015) investigate the effect of options on approximate planning algorithms. Their analysis considers options that use $\varepsilon$-optimal policies and are guaranteed to run for a large number of time steps, ensuring that few mistakes are made by each option. The key result of Mann et al. (2015) characterizes the convergence rate of value iteration using these kinds of options, with this rate depending directly on the sub-optimality and length of the options. Lehnert et al. (2018) explore the impact of horizon length on representation of value functions. They find that the value loss of any policy that optimizes with respect to an artificially-short horizon can achieve value similar to that of policies that take into account the full horizon, building on the results of Jiang et al. (2015b). Similar analysis is conducted in the case of well connected subsets of states, paralleling our treatment of state abstractions here.

**State-Action Abstraction.** Together, state and action abstractions can distill complex problems into simple ones (Jonsson and Barto, 2001; Ciosek and Silver, 2015). One popular approach builds around MDP homomorphisms (Ravindran and Barto, 2002, 2003a,b, 2004; Ravindran, 2003). MDP homomorphisms compress the MDP by collapsing state-action pairs that can be treated as equivalent. The recent work by Ma-jeed and Hutter (2019) extends this analysis to non-Markovian settings, proving the existence of several classes of value preserving homomorphisms. The key difference between their results and the analysis we present here is our close attachment to the options formalism. Indeed, as we will show, our framework for state-action abstractions can capture MDP homomorphisms in a sense.

Separately, Bai and Russell (2017) develop a Monte Carlo planning algorithm that incorporates state and action abstractions to efficiently solve the Partially Observable MDP (Kaelbling et al., 1998) induced by the abstractions. Theorem 1 from Bai and Russell (2017) shows that the value loss of their approach is bounded as a function of the state aggregation error (Hostetler et al., 2014), and Theorem 2 shows their algorithm converges to a recursively optimal policy for given state-action abstractions. Our results are closely related, but capture broad classes of state-action abstraction (Theorem 1) and target global optimality, rather than recursive optimality (Theorem 3).

**Hierarchical Abstraction.** Just as state and action abstractions enable a coarser view of a decision-making problem through a single lens, many approaches accommodate abstraction at multiple levels of granularity (Dayan and Hinton, 1993; Wiering and Schmidhuber, 1997; Parr and Russell, 1998; Dietterich, 2000a; Barto and Mahadevan, 2003; Jong and
Stone, 2008; Konidaris, 2016; Bai and Russell, 2017; Konidaris et al., 2018; Levy et al., 2019). Most recently, Nachum et al. (2019) introduce a scheme for preserving near-optimal behavior in hierarchical abstractions. Their main result studies the kinds of multi-step state visitation distributions induced by different hierarchies. In particular, they show that hierarchies that can induce a sufficiently rich space of k-step state distributions can represent near-optimal behavior, too. Indeed, this represents the strongest existing result for how to construct value-preserving hierarchies based on approximate knowledge.

3 ANALYSIS: $\phi$-RELATIVE OPTIONS

We incorporate state and action abstraction into RL as follows. When the environment transitions to a new state $s$, the agent processes $s$ via $\phi$ yielding the abstract state, $s_\phi$. Then, the agent chooses an option from among those that initiate in $s_\phi$, and follow the chosen option’s policy until termination, where this process repeats. In this way, an RL agent can reason in terms of abstract state and action alone, without knowing the true state or action space.

To analyze the value loss of these joint abstractions, we first introduce $\phi$-relative options, a novel means of combining state abstractions with options.

**Definition 4 ($\phi$-Relative Option):** For a given $\phi$, an option is said to be $\phi$-relative if and only if there is some $s_\phi \in S_\phi$ such that, for all $s \in S$:

$$ I_\phi(s) \equiv s \in s_\phi, \quad \beta_\phi(s) \equiv s \notin s_\phi, \quad \pi_\phi \in \Pi_{s_\phi}, \quad (1) $$

where $\Pi_{s_\phi} : \{ s' \mid \phi(s') = s_\phi \} \rightarrow \Delta(A)$ is the set of ground policies defined over states in $s_\phi$, and $s \in s_\phi$ is shorthand for $s \in \{ s' \mid \phi(s') = s_\phi, s' \in S \}$. We denote $O_\phi$ as any non-empty set that 1) contains only $\phi$-relative options, and 2) contains at least one option that initiates in each $s_\phi \in S_\phi$.

Intuitively, these options initiate in exactly one abstract state and terminate when the option policy leaves the abstract state. We henceforth denote $(\phi, O_\phi)$ as a state abstraction paired with a set of $\phi$-relative options.

**Example.** Consider the classical Four Rooms domain pictured in Figure 1a. Suppose further that the state abstraction $\phi$ turns each room into an abstract state. Then any $\phi$-relative option in this domain would be one that initiates anywhere in one of the rooms and terminates as soon as the agent leaves that room. The only degree of flexibility in grounding a set of $\phi$-relative options for the given $\phi$, then, is which policies are associated with each option, and how many options are available in each abstract state. If, for instance, the optimal policy $\pi^*$ were chosen for an option in the top right room, but the uniform random policy were available everywhere else, how might that impact the overall suboptimality of the policies induced by the abstraction? Our main result (Theorem 1) clarifies the precise conditions under which $\varepsilon$-optimal policies are representable under different abstractions.

To analyze the value loss of these pairs, we first show that any $(\phi, O_\phi)$ gives rise to an abstract policy over $S_\phi$ and $O_\phi$ that induces a unique policy in the original MDP (over the entire state space). Critically, this property does not hold for arbitrary options due to their semi-Markovian nature.

Proofs of all introduced Theorems and Remarks are presented in Appendix A.

**Remark 1.** Every deterministic policy defined over abstract states and $\phi$-relative options, $\pi_{O_\phi} : S_\phi \rightarrow O_\phi$, induces a unique Markov policy in the ground MDP, $\pi^{\triangledown}_{O_\phi} : S \rightarrow \Delta(A)$. We let $\Pi_{O_\phi}$ denote the set of abstract policies representable by the pair $(\phi, O_\phi)$, and $\Pi^{\triangledown}_{O_\phi}$ denote the corresponding set of policies in the original MDP.

This remark gives us a means of translating a policy over $\phi$-relative options into a policy over the original state and action space, $S$ and $A$. Consequently, we can define the value loss associated with a set of options paired with a state abstraction: every $(\phi, O_\phi)$ pair yields a set of policies in the original MDP, $\Pi^{\triangledown}_{O_\phi}$. We define the value loss of $(\phi, O_\phi)$ as the value loss of the best policy in this set.

![Diagram](image-url)
Definition 5 ((φ, O_φ)-Value Loss): The value loss of (φ, O_φ) is the smallest degree of sub-optimality achievable:

\[ L(φ, O_φ) := \min_{π_{φ,o} ∈ H_{O_φ}} \left\| V^* - V^π_{φ,o} \right\|_∞. \]  

Note that this notion of value loss is not well defined for options in general, since they induce a semi-MDP; there is no well-formed ground value function of a policy over options, but rather, a semi-Markov value function. As a simple illustration, consider a ground state s_g, two options o_1 and o_2 (either of which could be executing in s_g), and a policy π_{φ,o} over abstract states and options. It could be that o_1 or o_2 is currently executing when s_g is entered or that either option has just terminated, requiring π_{φ,o} to select a new option. Each of these three cases induces a distinct value V^{π_{φ,o}}(s_g) which is then difficult to distill into a single ground value function. This is a key reason to restrict attention to φ-relative options, each of which retains structure that couples with the corresponding state abstraction φ to yield value functions in the ground MDP.

3.1 Main Results

We next show how different classes of φ-relative options can represent near-optimal policies. We define an option class by a predicate λ : O_φ → {0, 1}, and say that a set of φ-relative options O_φ belongs to the class O_φ,λ if and only if λ(O_φ) = 1.

We begin by summarizing the four new φ-relative option classes, drawing inspiration from other forms of abstraction (Dean and Givan, 1997; Li et al., 2006; Jiang et al., 2015a; Abel et al., 2016; Ravindran and Barto, 2004; Nachum et al., 2019). For each class, we will refer to the optimal option in s_φ, o_φ^*, as the φ-relative option which initiates in s_φ and executes π^* until termination. These classes were chosen as they closely parallel existing properties studied in the literature. The four classes are as follows:

1. Similar Q*-Functions: In each s_φ, there is at least one option o that has similar Q* to o_φ^*.
2. Similar Models: In each s_φ, there is at least one option o that has a similar multi-time model (Precup and Sutton, 1998) to o_φ^*.
3. Similar k-Step Distributions: In each s_φ, there is at least one option o that has a similar k-step termination state distribution to o_φ^*\', based on the hierarchical construction introduced by Nachum et al. (2019). Loss bounds will only apply to goal-based MDPs.
4. Approximate MDP Homomorphisms: We show that any deterministic π_{O_φ} can encode an MDP homomorphism. The MDP homomorphism option class is defined by a guarantee on the quality of the resulting homomorphism.

Our main result establishes the bounded value loss of pairs (φ, O_φ) where O_φ belongs to any of these four classes, and the size of the bound depends on the degree of approximation (ε_Q, ε_R, ε_T; and ε_r, ε_p).

Hence:

\[ \lambda(O_φ) \equiv \forall s_φ \in S_φ : \exists o_φ \in O_φ : \max_{s \in s_φ} \left\| Q_{s_φ}^*(s, o_φ^*) - Q_{s_φ}^*(s, o) \right\| ≤ ε_Q, \]

where

\[ Q_{s_φ}^*(s, o) := R(s, π_o(s)) + γ \sum_{s′ \in S} T(s′ \mid s, π_o(s)) \left( I(s′ ∈ s_φ)Q_{s_φ}^*(s′, o) + I(s′ /∈ s_φ)V^*(s′) \right), \]

This Q-function describes the expected return of starting in state s, executing a φ-relative option o until leaving φ(s), then following the optimal policy thereafter.
More generally, this class of $\phi, O_\phi$ pairs captures all cases where each abstract state has at least one option that is useful. Note that the identity state abstraction paired with the degenerate set of options that exactly encodes the execution of each primitive action will necessarily be an instance of this class.

**Similar Models ($O_{\phi,M}$):** The $\varepsilon$-similar $T$ and $R$ predicate defines an option class where:

$$\lambda(O_\phi) \equiv \forall s_a, s_\phi \exists o_\phi \in O_\phi :$$

$$||T_{s_\phi, o_\phi}^s - T_{s_\phi, o_\phi}^{s'}||_\infty \leq \varepsilon_T$$

and

$$||R_{s_\phi, o_\phi} - R_{s_\phi, o_\phi}^{s'}||_\infty \leq \varepsilon_R,$$

where $R_{s_\phi, o_\phi}$ and $T_{s_\phi, o_\phi}^{s'}$ are shorthand for the reward model and multi-time model of Sutton et al. (1999). Roughly, this class states that there is at least one option in each abstract state that behaves similarly to the optimal option in that abstract state, $o_\phi^*$, throughout its execution in the abstract state.

We next derive two classes of $\phi$-relative options based on abstraction formalisms from existing literature. The first is based on the hierarchical construction introduced by Nachum et al. (2019), while the second shows that $\phi$-relative options can describe an MDP homomorphism (Ravindran and Barto, 2004).

**Similar k-Step Distributions ($O_{\phi,T}$):** Let $P(s', k \mid s, o)$ denote the probability of option $o$ terminating in $s'$ after $k$ steps, given that it initiated in $s$. We define this class by the following predicate:

$$\lambda(O_\phi) \equiv \forall s_a, s_\phi \exists o_\phi \in O_\phi :$$

$$\max_{s \in s_a, s' \in S} |P(s', k \mid s, o_\phi^*) - P(s', k \mid s, o)| \leq \tau.$$  

For the next class definition we first define the one-step abstract transition and reward functions for a $\phi$-relative option $o$:

$$T_\phi(s_\phi \mid s_\phi, o) = \sum_{s \in s_a} w(s) \sum_{s' \in s_\phi^*} T(s' \mid s, \pi_o(s)), \quad (11)$$

and

$$R_\phi(s_\phi, o) = \sum_{s \in s_a} w(s) R(s, \pi_o(s)), \quad (12)$$

where $w(s)$ is any valid weighting function with $\sum_{s \in s_a} w(s) = 1$. Next, we define the quantities of Ravindran and Barto (2004), related to the simulation lemma of Kearns and Singh (2002):

$$K_p = \max_{s \in S, a \in A}$$

$$\sum_{s \in s_a} \sum_{s' \in s_\phi} T(s' \mid s, a) - T_\phi(s_\phi \mid \phi(s), \pi_{O_\phi}(\phi(s))),$$

and

$$K_r = \max_{s \in S, a \in A} |R(s, a) - R_\phi(\phi(s), \pi_{O_\phi}(\phi(s)))|.$$  

These capture the maximum discrepancy between the model of the ground MDP and the model of the induced abstract MDP defined according to $(\phi, O_\phi)$. Then, we define the class as follows.

**Approximate MDP Homomorphisms ($O_{\phi,H}$):** This class requires that $O_\phi$ represents policies that induce good approximate homomorphisms:

$$\lambda(O_\phi) \equiv \forall \pi_{O_\phi} \in \Pi_{O_\phi} : K_p \leq \varepsilon_p$$

and

$$K_r \leq \varepsilon_r.$$ 

These four classes constitute four sufficient conditions for $(\phi, O_\phi)$ pairs to yield bounded value loss. It is useful, however, to identify not just sufficient conditions, but also necessary. To this end, we next establish one necessary condition of all (globally) value preserving $(\phi, O_\phi)$ classes.

**Theorem 2.** For any $(\phi, O_\phi)$ pair with $L(\phi, O_\phi) \leq \eta$, there exists at least one option per abstract state that is $\eta$-optimal in Q-value. Precisely, if $L(\phi, O_\phi) \leq \eta$, then:

$$\forall s_a \in S_a \forall s_\phi \exists o_\phi \in O_\phi : Q_{s_a}^*(s_\phi, o_\phi^*) - Q_{s_a}^*(s, o) \leq \eta. \quad (14)$$

This theorem tells us that for any agent acting using our joint abstraction framework, if there exists an abstract state for which there is not an $\eta$-optimal option, then the agent cannot represent a globally near-optimal policy.

### 3.2 Experiment

We next conduct a simple experiment to corroborate the findings of our analysis, and to test whether value preserving options enable simple RL algorithms to find near-optimal policies. The experiment illustrates an important property of one of the introduced $\phi, O_\phi$ classes, and is organized as follows. We first construct a $\phi, O_\phi$ pair belonging to the $O_{\phi,H}$ class using dynamic programming. We give this pair to an instance of Double Q-Learning (Hasselt, 2010) with a varied sample budget $N$. The environment is a Four Rooms grid world MDP, with a single goal location in the top right and start location in the bottom left. The state abstraction $\phi$ maps each state into one of four abstract states, denoting each of the four rooms. We vary both the number of options added per abstract state ($|O_\phi|$) and the sample budget of Double Q ($N$), and present the value of the policy discovered by the final episode for each setting of $|O_\phi|$ and $N$.

Results are presented in Figure 2a. First, note that with only one option per abstract state, Double Q can trivially find a near-optimal policy, even with a small sample budget. This aligns with our theory: the included options preserve value, and so any assignment
of options to abstract states will yield a near-optimal policy. In contrast, if randomly chosen options are used instead (shown in blue, labeled as $O_{\text{random}}$), the learning algorithm fails to find a good policy even with a high sample budget ($N = 1 \cdot 10^6$ was used). Second, we find that as the number of options increases, the added branching factor causes Double Q-Learning to find a lower value policy with the same number of samples. However, by Theorem 1 we know each set of options preserves value; as the sample budget increases we see that the value of the discovered policy tends toward optimal. In short, the $\phi, O_{\phi}$ pairs defined by Theorem 1 do in fact preserve value, but will also affect the sample budget required to find a good policy. We foresee the combination of value preserving abstractions with those that lower learning complexity (see recent work by Brunskill and Li (2014); Fruit et al. (2017)) as a key direction for future work.

We further visualize the learned value function of Q-Learning with and without $\phi$-relative options after the same sample budget, depicted in Figure 2b. Notably, since $\phi$-relative options update entire blocks of states, we see large regions of the state space with the same learned value function. Conversely, Q-Learning only tends to explore (and estimate the values of) a narrow region of the state space. The visual highlights this important qualitative difference between learning with and without abstractions. More experiments, details, and a link to the code are found in Appendix B.

4 HIERARCHICAL ABSTRACTION

We next present an extension of Theorem 1 that applies to hierarchies consisting of $(\phi, O_{\phi})$ pairs. We show the value loss compounds linearly if we construct a hierarchy using algorithms that generate a well-behaved $\phi$ and $O_{\phi}$. To do so, we require two definitions and additional notation (see Table 1 in Appendix A for a summary of hierarchy notation). We first define a hierarchy as $n$ sets of $(\phi, O_{\phi})$ pairs.

Definition 6 ($\phi, O_{\phi}$-Hierarchy): A depth $n$ hierarchy, denoted $H_n$, is a list of $n$ state abstractions, $\phi^{(n)}$, and a list of $n$ sets of $\phi$-relative options, $O_{\phi}^{(n)}$. The components $(T, \beta, \pi)$ of each of the $i$-th set of options, $O_{\phi,i}$ are defined over the $(i - 1)$-th abstract state space $S_{\phi,i-1} = \{\phi_{i-1}(\phi_{i-2}(\ldots \phi_1(s) \ldots)) | s \in S\}$.

We next introduce additional notation to refer to values, states, options, and policies at each level of the hierarchy. We denote $\pi_n : S_{\phi,n} \rightarrow O_{\phi,n}$ as the level $n$ policy encoded by the hierarchy, with $\Pi_n$ the space of all policies encoded in this way. We let $s_i := \phi^i(s) = \phi_i(\ldots \phi_1(s) \ldots)$, with $s$ a state in the ground MDP. We further denote $V_i$ as the $i$-th level’s value function, defined as follows for some ground state $s$:

$$V^\pi_i(s) := V^\pi_i(\phi^i(s)) = V^\pi_i(s_i) = \max_{o \in C_i} \left( R_i(s_i, o) + \sum_{s' \in S_i} T_i(s' | s_i, o) V^\pi_i(s') \right) , \quad (15)$$

where:

$$R_i(s_i, o) := \sum_{s_{i-1} \in S_i} w_i(s_{i-1}) R_{s_{i-1}, o} \quad (16)$$

$$T_i(s' | s_i, o) := \sum_{s_{i-1} \in S_i} \sum_{s_{i-1}' \in S_{i-1}} w_i(s_{i-1}) T_{s_{i-1}, o}$$

where again $R_{s,o}$ and $T_{s,o}$ are defined according to the multi-time model (Sutton et al., 1999), $s_i \in S_{\phi,i}$ is a
level \( i \) state resulting from \( \phi^i(s) \), and \( w_i \) is an aggregation weighting function for level \( i \). Note that \( V_0 \) is the ground value function, which we refer to as \( V \) for simplicity.

### 4.1 Hierarchy Analysis

We now extend Theorem 1 to hierarchies of arbitrary depth, building on two key observations. First, any policy \( \pi_n \) represented at the top level of a hierarchy \( H_n \) also has a unique Markov policy in the ground MDP, which we denote \( \pi_n^0 \) (in contrast to \( \pi_n^\uparrow \), which moves the level \( n \) policy to level \( n - 1 \)). We summarize this fact in the following remark:

**Remark 2.** Every deterministic policy \( \pi_i \) defined by the \( i \)-th level of a hierarchy, \( H_n \), induces a unique policy in the ground MDP, which we denote \( \pi_i^0 \).

To be precise, note that \( \pi_i^\uparrow \) specifies the level \( i \) policy \( \pi_i \) mapped into level \( \pi_{i-1} \), whereas \( \pi_i^0 \) refers to the policy at \( \pi_i \) mapped into \( \pi_0 \). The second key insight is that we can extend our same notion of value loss from \((\phi, \mathcal{O}_\phi)\) pairs to hierarchies, \( H_n \).

**Definition 7** \((H_n, \text{-Value Loss})\): The value loss of a depth \( n \) hierarchy \( H_n \) is the smallest degree of suboptimality across all policies representable at the top level of the hierarchy:

\[
L(H_n) := \min_{\pi_n \in H_n} \left\| V^* - V^{\pi_n^0} \right\|_{\infty}.
\]

This quantity denotes how suboptimal the best hierarchical policy is in the ground MDP. Therefore, the guarantee we present expresses a condition on global optimality rather than recursive or hierarchical optimality (Dietterich, 2000a).

We next show that there exist value-preserving hierarchies by bounding the above quantity for well-structured hierarchies. To prove this result, we require two assumptions.

**Assumption 1.** The value function is consistent throughout the hierarchy. That is, for every level of the hierarchy \( i \in [1 : n] \), for any policy \( \pi_i \) over states \( S_{\phi, i} \) and options \( \mathcal{O}_{\phi, i} \), there is a small \( \kappa \in \mathbb{R}_{\geq 0} \) such that:

\[
\max_{s \in S} \left| V_{i+1}^{\pi_i} (\phi^{-1} (s)) - V_i^{\pi_i} (\phi (s)) \right| \leq \kappa
\]

**Assumption 2.** Subsequent levels of the hierarchy can represent policies similar in value to the best policy at the previous level. That is, for every \( i \in [1 : n - 1] \), letting \( \pi_i^0 = \arg \min_{\pi_i \in \Pi_i} \left\| V_0 - V_0^{\pi_i^0} \right\|_{\infty} \), there is a small \( \ell \in \mathbb{R}_{\geq 0} \) such that:

\[
\min_{\pi_{i+1} \in \Pi_{i+1}} \left\| V_i^{\pi_i^0} - V_i^{\pi_{i+1}^0} \right\|_{\infty} \leq \ell.
\]

We strongly suspect that both assumptions are true given the right choice of state abstractions, options, and methods of constructing abstract MDPs. These two assumptions (along with Theorem 1) give rise to hierarchies that can represent near-optimal behavior. We present this fact through the following theorem:

**Theorem 3.** Consider two algorithms: 1) \( A_\phi \): given an MDP \( M \), outputs a \( \phi \), and 2) \( A_{\mathcal{O}_\phi} \): given \( M \) and a \( \phi \), outputs a set of options \( \mathcal{O} \) such that there are constants \( \kappa \) and \( \ell \) for which Assumption 1 and Assumption 2 are satisfied.

Then, by repeated application of \( A_\phi \) and \( A_{\mathcal{O}_\phi} \), we can construct a hierarchy of depth \( n \) such that:

\[
L(H_n) \leq n(\kappa + \ell).
\]

This theorem offers a clear path for extending the guarantees of \( \phi \)-relative options beyond the typical two-timescale setup observed in recent work (Bacon et al., 2017; Nachum et al., 2019) to fully realize the benefits of (multi-level) hierarchical abstraction (Levy et al., 2019). Moreover, both Assumption 1 and Assumption 2 are sufficient—together with \( \phi \)-relative options that satisfy Theorem 1—to construct a hierarchy with low value loss. We conclude that algorithms for leveraging hierarchies may want to explicitly search for structures that satisfy our assumptions: 1) value function smoothness up and down the hierarchy, and 2) policy richness at each level of the hierarchy.

## 5 DISCUSSION

We proved which state-action abstractions are guaranteed to preserve representation of high value policies. To do so, we introduced \( \phi \)-relative options, a simple but expressive formalism for combining state abstractions with options. Under this formalism, we proposed four classes of \( \phi \)-relative options with bounded value loss. Lastly, we proved that under mild conditions, pairs of state-action abstractions can be recursively combined to induce hierarchies that also possess near-optimality guarantees.

We take these results to serve as a concrete path toward principled abstraction discovery and use in RL. To realize this goal, abstractions need to both preserve good behavior, and lower sample complexity. Our results thus far focus on the representation of near-optimal policies, which is a key condition for ensuring agents can eventually behave well. However, learning agents also need to represent functions of intermediate quality as they learn. Thus, an important direction for future work is to clarify which kinds of abstractions ensure that near-optimal policies are easily learnable.
Acknowledgements

We would like to thank Cam Allen, Philip Amortila, Will Dabney, Mark Ho, George Konidaris, Ofir Nachum, Scott Niekum, Silviu Pitis, and Balaraman Ravindran for insightful discussions, and the anonymous reviewers for their thoughtful feedback. This work was supported by a grant from the DARPA L2M program and an ONR MURI grant.

References


Kamil Ciosek and David Silver. Value iteration with options and state aggregation. ICAPS Workshop on Planning and Learning, 2015.


Jean Harb, Pierre-Luc Bacon, Martin Klissarov, and Doina Precup. When waiting is not an option:
Value Preserving State-Action Abstractions


Sultan Javed Majeed and Marcus Hutter. Performance guarantees for homomorphisms beyond Markov de-


