APPROXIMATING THE OPTIMAL STATE ABSTRACTION

Abstract.

1. Background

1.1. **The Problem.**

Goal: planning and learning in large[a] state spaces.
Proposed Strategy: abstract the problem to simplify.

[a]TOTALLY MASSIVE

For example: Historically, we've relied on OO-MDPs a lot. There are other possible representations one might consider to collapse our representation of the real world. Objects seem like a reasonable choice given how object-oriented the human experience seems to be.

The main point is that compression helps, but we don't totally know how effective these things could be, and we don't know exactly which compressions to choose. In some sense, we're just picking compressions based on what we think might work.

Some questions to consider:

- Q: Is there a unified definition of state abstraction?
- Q: How do different abstractions relate to one another?
- Q: How can we select among abstraction schemes?
- Q: How does the solution to the abstracted MDP relate to the original MDP?
- Q: How should we be compressing? Does it change from domain to domain, task to task?
- Q: Can we learn these abstractions? How much data do they require?

Immediate question: How should we compress? Any ideas? Does it change depending on the domain? The task? Are there other things we can use compression for?

So it turns out there have been a huge number of attempts at this:

- Dynamic Programming Aggregation
- Stochastic DP with factored representations
- Model reduction techniques
- Model minimization...
- Abstraction Selection...
- State Abstraction Selection from...
- Proto-value functions
- Selecting the state-representation in reinforcement learning
- Optimal regret bounds for selecting the...
- RL with selective perception
- SMDP homomorphisms: an algebraic approach.

- Selecting near optimal approximate state...

1.2. **Unified Theory Paper.** Michael, Tom Walsh, Lihong Li wrote a paper called "Towards a Unified Theory of State Abstraction for MDPs" [**?**] that surveys these approaches, and puts them into a single framework.

The high level is that there are roughly 5 types of abstractions:

(1) Model-irrelevance: ϕ_{model} :

$$\forall_s \forall_a \phi_{\text{model}}(s_1) = \phi_{\text{model}}(s_2) \rightarrow \mathcal{R}(s_1, a) = \mathcal{R}(s_2, a) \wedge \sum_{s' \in \phi^{-1}_{\text{model}}(s)} \Pr(s' \mid s_1, a) = \sum_{s' \in \phi^{-1}_{\text{model}}(s)} \Pr(s' \mid s_2, a)$$

(2) Q^π-irrelevance: ϕ_{Q^π} :

$$\forall_\pi \phi_{Q^\pi}(s_1) = \phi_{Q^\pi}(s_2) \rightarrow \forall_a Q^\pi(s_1, a) = Q^\pi(s_2, a)$$

(3) Q^*-irrelevance: ϕ_{Q^*} :

$$\phi^*_Q(s_1) = \phi^*_Q(s_2) \rightarrow \forall_a Q^*(s_1, a) = Q^*(s_2, a)$$

(4) a^*-irrelevance: ϕ_{a^*}:

$$\phi_{a^*}(s_1) = \phi^*_a(s_2) \rightarrow Q^*(s_1, a^*) = \max_a Q^*(s_1, a) = \max_a Q^*(s_2, a) = Q^*(s_2, a)$$

(5) π^*-irrelevance: ϕ_{π^*}:

$$\phi_{\pi^*}(s_1) = \phi_{\pi^*}(s_2) \rightarrow \left(Q^*(s_1, a^*) = \max_a Q^*(s_1, a) \wedge Q^*(s_2, a^*) = \max_a Q^*(s_2, a) \right)$$

One note: I haven't thought about this too much, but I'm pretty sure that throwing in abstracted actions like Options fits into this framework neatly – suppose we're in an SMDP and that an option is just an action. Then we can abstract the state space based on where subgoals are/aren't satisfied. Seems like the right sort of result.

Other results from the paper:

Theorem 3: With abstractions ϕ_{model}, ϕ_{Q^π}, ϕ_{Q^*}, and ϕ_{a^*}, the optimal abstract policy $\bar{\pi}^*$ is optimal in the ground MDP.

Theorem 4.1: Q-Learning with abstractions ϕ_{model}, ϕ_{Q^π}, and ϕ_{Q^*}, converges to the optimal state-action value function in the ground MDP

Theorem 4.2: Q-Learning with abstraction ϕ_{a^*} does not necessarily converge.

Theorem 4.3: Q-Learning with abstraction ϕ_{π^*} can converge to an action-value function whose greedy-policy is suboptimal in the ground MDP.

In a follow up paper, they have a distribution on MDPs, sample some training MDPs to infer the optimal state abstraction, and use it to solve a test MDP from the same distribution.

2. Motivation

Basic Point: State abstraction for planning, learning, possibly bandits could be EPIC.

Result 1: Arbitrary reduction in Sample Complexity for a particular MDP.

Result 2: More general result about sample complexity reduction for MDPs of a certain type

The Question: Assuming a given MDP has property set X, can we say anything about the possible reduction in sample complexity using a state abstraction function.

Possible other Result: Exploration.

3. Elephants in the Room

(1) Finding states with *exactly* the same Q values, or optimal action and action value, or same model, is rare!

(2) Can't capture temporal abstraction.

Proposal: Resolve (1) and (2), so that the nice results from Section 2 can be realized. The next two sections talk about how we might go about doing that.

---

4. New Proposal 1: Approximate Abstraction

For each of the following four cases:

$$\phi(s_1) = \phi(s_2) \rightarrow \forall_{s,a} : T(s \mid s_1, a) = T(s \mid s_2, a) \wedge \forall_a : R(s_1, a) = R(s_2, a) \tag{1}$$

$$\phi(s_1) = \phi(s_2) \rightarrow \forall_{s,a} : T(s \mid s_i, a) = T(s \mid s_j, a) \tag{2}$$

$$\phi(s_1) = \phi(s_2) \rightarrow \forall_a : Q^*(s_i, a) = Q^*(s_j, a) \tag{3}$$

$$\phi(s_1) = \phi(s_2) \rightarrow \arg\max_a Q^*(s_i, a) = \arg\max_a Q^*(s_j, a) \wedge \max_a Q^*(s_i, a) = \max_a Q^*(s_j, a) \tag{4}$$

Our strategy: relax the equality condition, instead consider the approximate case, e.g.:

$$\phi(s_1) = \phi(s_2) \rightarrow \forall_{s,a} : |T(s \mid s_i, a) - T(s \mid s_j, a)| \leq \epsilon \tag{5}$$

Sort of like the Simulation Lemma from E^3.

4.1. **The Question:** Suppose we're given the optimal policy in the abstract MDP under one of the above approximate abstractions, e.g. $\pi^*_{\phi_{T,\epsilon}}$. What can be said about the potential optimality of this policy in the original ground MDP? (for each of the four approximate abstractions).

4.2. **Followup Question:** Why *won't* this work for other types of abstractions?

- Just using state variables...
- Just Reward function at a state...
- Just Value of state...
- Any combination of these...

Results about which abstractions *don't* work. Note: Already have counter examples for the above 3, would be nice to get more general results.

The result is that we identify which criteria are at the core of making state abstraction *useful*.

5. Results

Lemma 1: Given two MDPs, M_1 and M_2 which differ only by T and R, if $\forall_{s,a}|T_1(s,a,\cdot) - T_2(s,a,\cdot)| \leq 2\beta$ and $\forall_{s,a}|R_1(s,a) - R_2(s,a)| \leq \alpha$ then for some fixed policy π then $\forall_{s,a}|Q_1^\pi(s,a) - Q_2^\pi(s,a)| \leq \frac{\alpha v_{max}}{C(1-\gamma)} = \epsilon$ (Simulation Lemma)

Call the ground truth MDP M_G, the abstract MDP $M_A = \phi(M_G)$.[1] Suppose that the agent plans in M_A. We would like to bound how well it is *really* performing in M_G based on how it performs in M_A. Call the decompressed ground state of M_A, $M_{G'}$, the noisy ground MDP.

Lemma 2: Under the model approximate abstraction scheme, ϕ_{model}, the Q values under any policy in M_A will be within ϵ of Q values under the same policy in M_G.

Proof: Recall our definition of model approximate abstraction:

$$\forall_{s_1,s_2}\phi(s_1) = \phi(s_2) \rightarrow \forall_{s,a}|T(s,a,s_1) - T(s,a,s_2)| \leq 2\beta \tag{6}$$

Sim Lemma says we need T and R to be within $2\beta = \alpha$.

We have our ground MDP: M_G, abstract MDP, M_A, and noisy ground MDP, $M_{G'}$.

Using a given state abstraction function ϕ subject to Equation 6, the difference between T_G and $T_{G'}$.

Sim lemma goes through (cause same for R).

Lemma 3: Under the optimal Q function approximate abstraction, $\forall_s : |Q_A^*(s,.) - Q_G^*(s,.)| \leq \epsilon$

Since for each state, the most you can screw up is by ϵ, since if you don't take $\arg\max_a Q^*(s,a)$, then one of the other actions Q functions must have jumped up by at most ϵ, but it's fine because then you're doing no worse then ϵ. It can't have jump mored, because otherwise you wouldn't have compressed on it.

[1] Sloppy notation – really ϕ is only applied to S.

6. New Proposal 2: Temporal Abstraction

In light of the above negative results (regarding which abstractions are useful), that state abstraction needs to be defined with respect to some sort of properties about actions, suppose a set of temporally extended actions are included in the action set.

Now, abstractions become temporally extended.

What sort of thing do we want to prove here? Abstractions become temporal if actions are temporally extended? Without temporal extension, forces $T(s, a, s')$ and $R(s, a)$ to do weird things?

Q: Can we treat temporally extended actions in a general enough way to prove things about all possible versions of TEAs, or do we need to do separate proofs for Options/Macroactions?

7. Other Low Hanging Fruit Results

7.1. **Result 3: Collapsing on just** *state variables*, **or** *reward*, **or** *value* **is insufficient to preserve any kind of optimality (including ϵ-optimality.**

Game Plan

First, get results about the approximate state abstraction for each of the 4 under consideration.

Second, investigate temporally extended actions w/ abstraction.

Third, wrap up low hanging fruit results, investigate more general versions of these claims (beyond $\exists MDPs.t. \ldots.$).

8. Conclusion

Future Work:

- Learning ϕ
- Connection to AMDPs (e.g. consider $\Omega : \langle S, A, R, T, \gamma \rangle \mapsto \langle S', A', R', T', \gamma' \rangle$
- Connection to teaching.
  - AMDPs seem well suited to hierarchical teaching Carl-style.
  - Problem: once solved for optimal policy in AMDP already solved in low level state. Probably want to set up teaching to avoid this.