Improving Off-Policy Hierarchical Reinforcement Learning in Soar

Mitchell Keith Bloch

University of Michigan
2260 Hayward Street
Ann Arbor, MI. 48109-2121
bazald@umich.edu

June 15-17, 2011
Outline

1 Background

2 Hierarchical Reinforcement Learning

3 Soar-RL

4 Nuggets and Coal

5 References
Reinforcement Learning

- The Reinforcement Learning (RL) problem – learn to maximize the expected discounted return from any reachable state
  - More simply – learn the optimal choice of action from each state
- Environment models can help, but are not always desirable
- SARSA($\lambda$) and Q($\lambda$) are popular model-free RL algorithms

![Figure: Temporal Difference (TD) Backup for a Q-Value](chart.png)
SARSA($\lambda$) is on-policy – learning about policy being followed
- Incorporates expected return of selected next action
- Optimizing the current policy

Q($\lambda$) is off-policy – not learning about policy being followed
- Incorporates expected return of best available next action
- Optimizing the optimal policy

In context of HRL – learning off-policy enables all-goals updating
- Learn about multiple goals concurrently
On/Off-Policy Cliff-Walking Domain

Exploration requires choosing non-greedy actions (occasionally going off the edge of the cliff)

On-Policy converges indirectly to the ultimately optimal policy

Figure: An on-policy agent with high exploration steers clear of the cliff
On/Off-Policy Cliff-Walking Domain

Exploration requires choosing non-greedy actions (occasionally going off the edge of the cliff)

On-Policy converges indirectly to the ultimately optimal policy

Figure: An on-policy agent with moderate exploration stays closer to the cliff
On/Off-Policy Cliff-Walking Domain

Exploration requires choosing non-greedy actions
(occasionally going off the edge of the cliff)

On-Policy converges indirectly to the ultimately optimal policy

Figure: An on-policy agent with low exploration stays adjacent to the cliff
On/Off-Policy Cliff-Walking Domain

Exploration requires choosing non-greedy actions (occasionally going off the edge of the cliff)

Off-policy converges directly to the ultimately optimal policy

Figure: An off-policy agent stays adjacent to the cliff regardless of exploration
Outline

1. Background
2. Hierarchical Reinforcement Learning
3. Soar-RL
4. Nuggets and Coal
5. References
Bandit Task of Interest

- **A** – Reward 1 – Escape into tunnel with dragon
- **B** – Reward 10 – Fight less dangerous monster (depicted)
- **C** – Reward 100 – Escape into tunnel with treasure
Large or complex problems involving separable goals can be broken down hierarchically.

Decouples the problem of deciding which goal to achieve next from the problem of how to achieve it.

Enables state abstraction and goal reuse.
Exploration and Learning

- Can explore non-greedy actions within a goal
  - Must learn correctly in supergoals regardless
- Can explore subgoals with no chance of success
  - Must learn correctly in subgoals regardless
Exploring Non-Greedy Actions - The Setup

Why are non-greedy actions in subgoals problematic?

Group actions A and C in a subtask, “Escape”. The decision procedure becomes:

1. Fight (B) or Escape?
2. If Escape, then (A) or (C)?
Exploring Non-Greedy Actions - The Mistake

1. The true value of Escape is 100, once Escape is learned.
2. Exploration, required by convergence proofs, causes Escape to yield only 1 reward.
3. The initial decision can learn that Escape is worth only 1 point.

Point 3 is true even when learning with $Q(\lambda)$. 
The Mistake - Visualized

**Figure:** Mean cumulative suboptimality for Naive RL asymptotes at approximately -50 reward in the limit, regardless of cooling strategy. Fixed HRL achieves an optimal policy but does worse than Flat RL due to higher persistent exploration: $(1 - \varepsilon)^2 < 1 - \varepsilon$
The true value of Escape is 100, once Escape is learned.

Exploration, required by convergence proofs, causes Escape to yield only 1 reward.

The initial decision can learn that Escape is worth only 1 point. This is not what we expect when learning off-policy!
The true value of Escape is 100, once Escape is learned.

Exploration, required by convergence proofs, causes Escape to yield only 1 reward.

The initial decision can learn that Escape is worth only 1

Point 3 is true even when learning with Q(\(\lambda\)).

This is not what we expect when learning off-policy!

**Conclusion:** Learning must be blocked by exploration in subgoals.
Hierarchical Credit Assignment

When does a goal bear responsibility for reward received?

- **On-Policy?** – Goal is attainable when selected by supergoals
- **Off-Policy?** – Additionally, all subgoals choose greedily
1 Background

2 Hierarchical Reinforcement Learning

3 Soar-RL

4 Nuggets and Coal

5 References
Soar-RL

- Implements RL using numeric preferences and the RL link
  - Actually, one RL link per goal for correct hierarchical credit assignment
- Supports both SARSA(\(\lambda\)) and Q(\(\lambda\)) [Nason and Laird, 2004; Derbinsky et al., 2009]
- Implements HRL using operator no-change impasses and multiple RL links
Recommendation 1: Exploration in Subgoals

When learning off-policy, TD updates must be blocked and eligibility traces must be cleared.

Intra-option learning [Sutton and Precup, 1998] and (G)TSDT [Bloch, 2011b,a] can improve this situation somewhat.
It is necessary to pursue a goal until success or failure for Soar-RL to learn in the context of HRL, but this commitment is not integral to Soar.

Supporting intra-option learning [Sutton and Precup, 1998] and (G)TSDT [Bloch, 2011b,a] would enable learning in cases where this commitment is not desired.
Recommendation 2: Operator No-Change Impasses

Learning on-policy or off-policy, terminal reward should be backed up as a goal retracts \textbf{iff} the impasse resolves normally.

A supergoal retracting should prevent TD updates.
Outline

1. Background
2. Hierarchical Reinforcement Learning
3. Soar-RL
4. Nuggets and Coal
5. References
Nuggets:

- Identified conditions under which HRL fails to work as expected
- Modified HSMQ [Dietterich, 2000] and Intra-option learning [Sutton and Precup, 1998], resulting in what I believe to be the first off-policy TD methods to converge reliably in model-free HRL systems
- Created new traces to improve performance over HSMQ and Intra-option learning [Bloch, 2011b,a] given the new constraints

Coal:

- No formal convergence proofs provided
- Not formally addressed function approximation (yet)
Outline

1. Background
2. Hierarchical Reinforcement Learning
3. Soar-RL
4. Nuggets and Coal
5. References


