Sample-Based Methods for Continuous Action Markov Decision Processes

Chris Mansley
Ari Weinstein
Michael Littman
Rutgers Unversity

From Learning to Planning

Bellman Equation

$$V(s) = \max_a \left(R(s,a) + \gamma \sum_{s'} T(s,a,s')V(s')\right)$$

From Learning to Planning

Bellman Equation

$$V(s) = \max_a (R(s,a) + \gamma \sum_{s'} T(s,a,s')V(s'))$$

Continuous State Space

Standard machine learning approaches to function approximation have proven successful!

From Learning to Planning

Bellman Equation

$$V(s) = \max_{a}\left(R(s,a) + \gamma \sum_{s'} T(s,a,s')V(s')\right)$$

$\boxed{\text{Continuous Action Space}}$

Very little work addressing how to evaluate the maximum

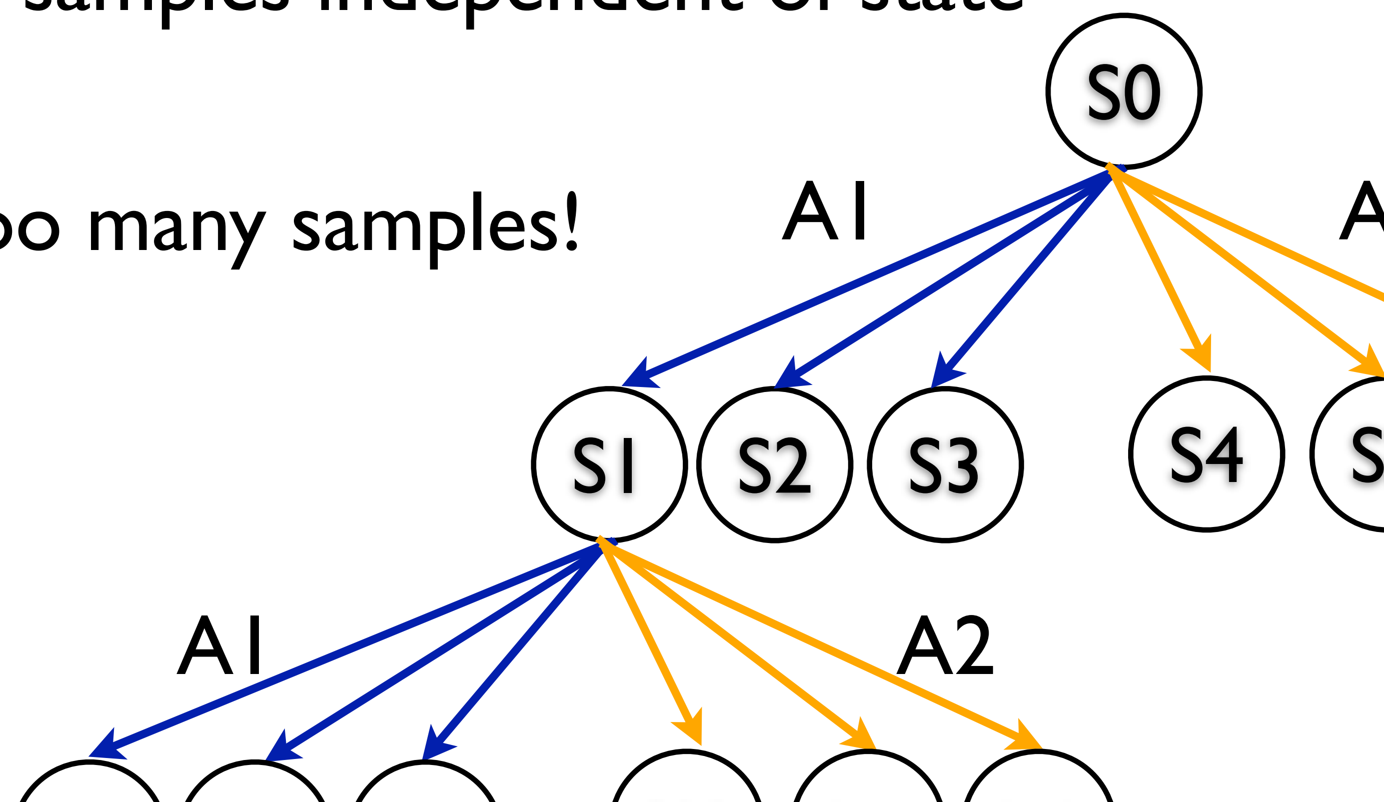
$\boxed{\text{Continuous State Space}}$

Standard machine learning approaches to function approximation have proven successful!

Sparse Sampling

[Kearns, et al 1999]

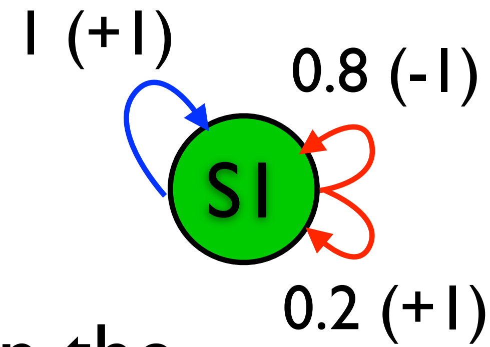
- An epsilon-optimal planning algorithm for discounted MDPs.
- Number of samples independent of state space size!
- Requires too many samples!



Can we use ideas from the exploration/exploitation problem to better direct our search?

UCB

[Auer, et al 2002]



- An algorithm for efficient learning in the bandit domain
- Fixed number of discrete actions with bounded support
- Choose an arm greedily according to the following rule:

$$\widehat{\mu_i} + \sqrt{\frac{2 \ln n}{n_i}}$$

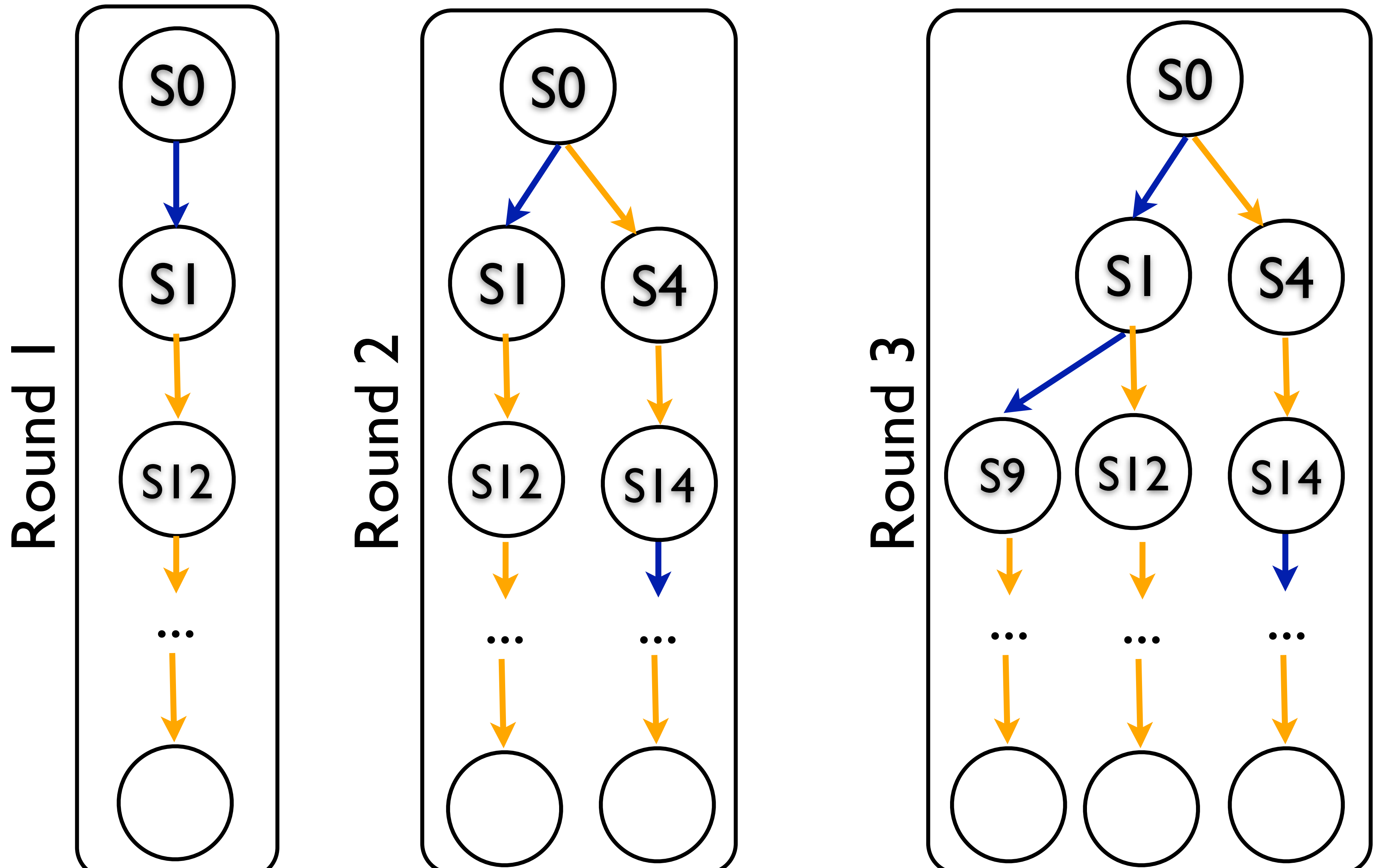
UCT

[Kocsis, Szepesvári 2006]

- Upper Confidence applied to Trees
- Takes the UCB algorithm and extends it to the full MDP domain
- Build a tree similar to SS, but instead of doing a breadth first search perform a depth first search directed by a UCB algorithm at each node

UCT, cont...

[Kocsis, Szepesvári 2006]



HOO

[Bubeck, et al 2008]

- UCT is still restricted to discrete states and actions
- HOO (hierarchical optimistic optimization) provides similar guarantees to UCB in "well-behaved" continuous bandit problems
- The idea is simple, divide the action space up (similar to a KD-tree), keep track of returns in these volumes, provide exploration bonuses for both number of samples and size of each subdivision

HOO, cont...

[Bubeck, et al 2008]

- Choose an arm greedily with respect to the following:

$$\widehat{\mu_i} + \sqrt{\frac{2\ln n}{n_i}} + v_1\rho^h$$

- Very similar to UCB except the spatial term at the end
- The intuition is that arms with large volumes and few samples are unknown, but small volumes and lots of samples are well known

HOO, cont...

[Bubeck, et al 2008]

- Choose an arm greedily with respect to the following:

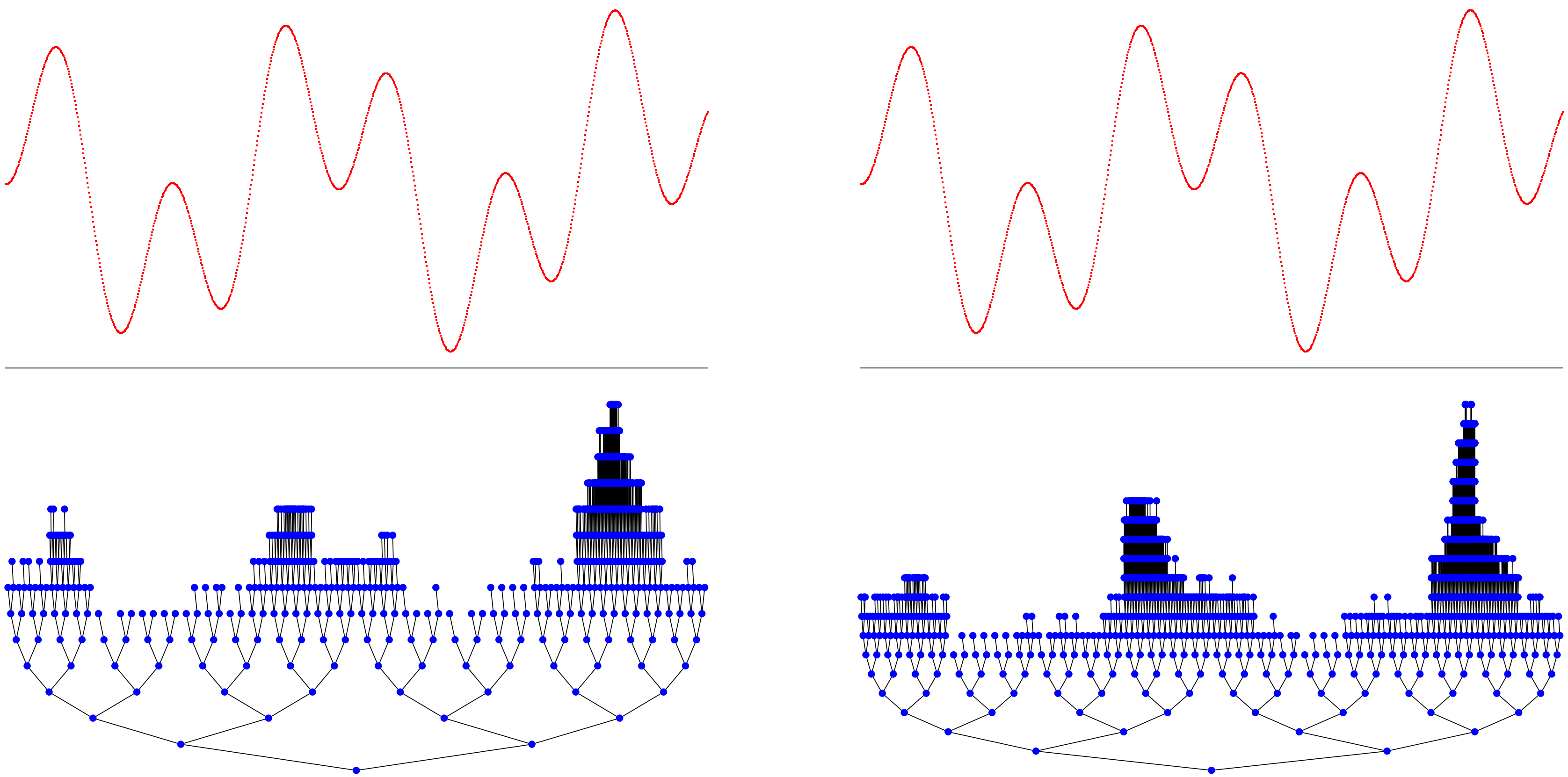
$$\widehat{\mu_i} + \sqrt{\frac{2\ln n}{n_i}} + \cancel{v_1 \rho^h} \text{ diam(i)}$$

- Very similar to UCB except the spatial term at the end

- The intuition is that arms with large volumes and few samples are unknown, but small volumes and lots of samples are well known

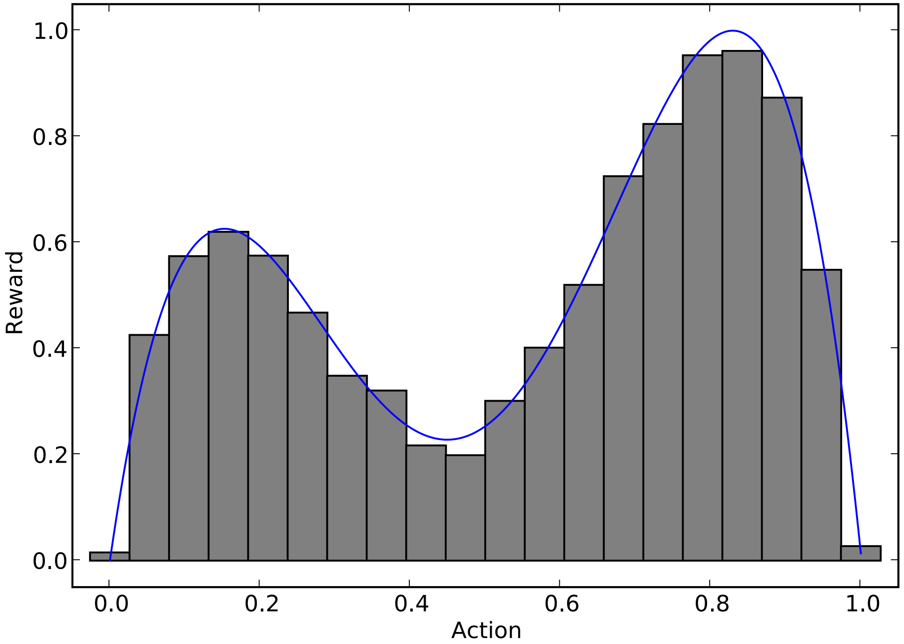
HOO, cont...

[Bubeck, et al 2008]

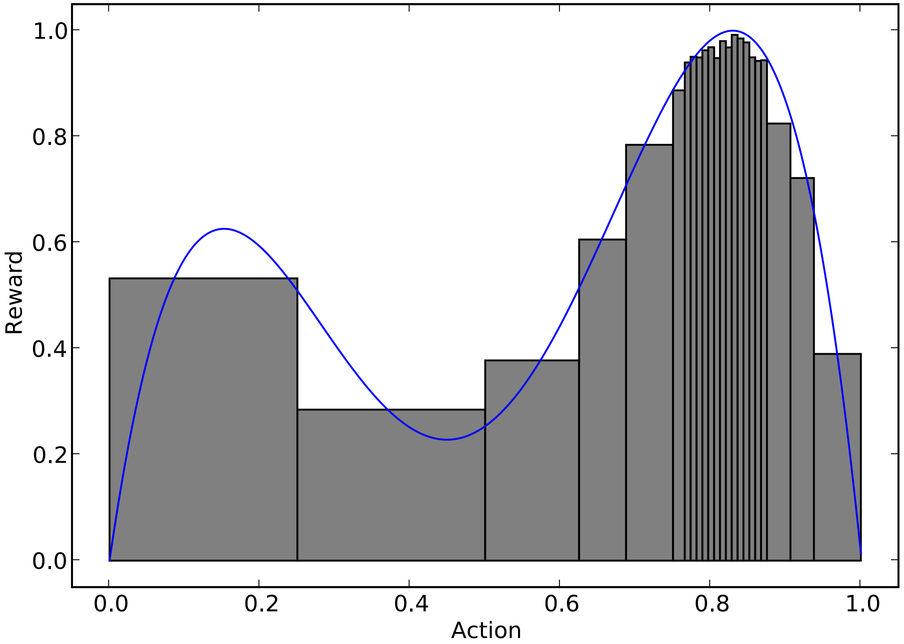


Thanks to Remi Munos

UCB vs HOO



1.0
0.8
0.6
0.4
0.2
0.0
Reward
0.0
0.2
0.4
0.6
0.8
1.0
Action

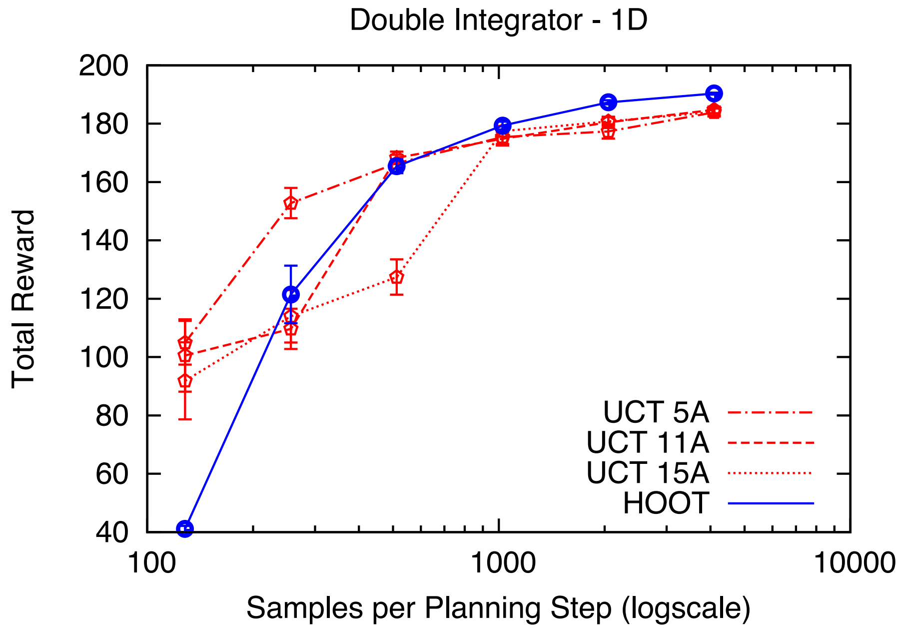


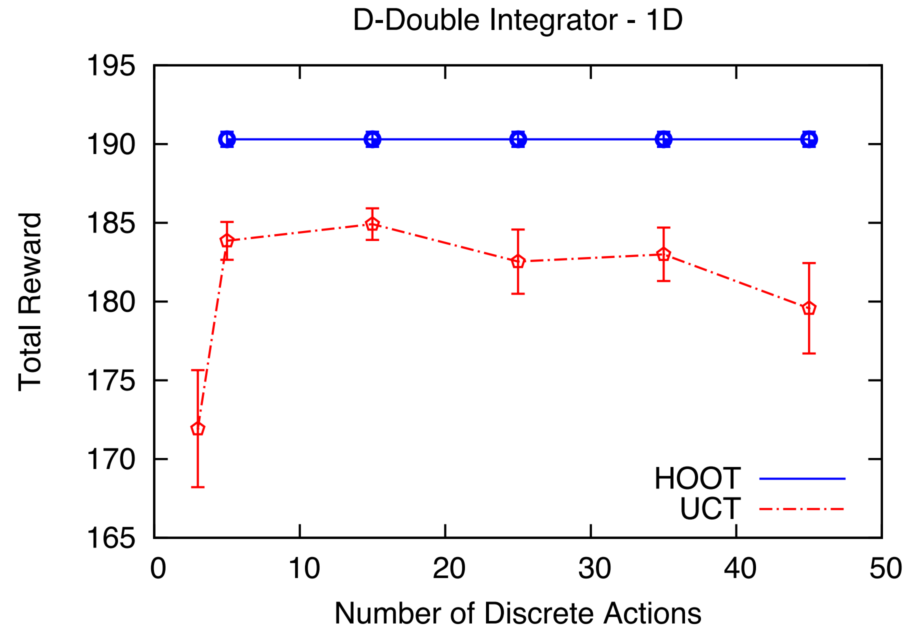
1.0
0.8
0.6
0.4
0.2
0.0
Reward
0.0
0.2
0.4
0.6
0.8
1.0
Action

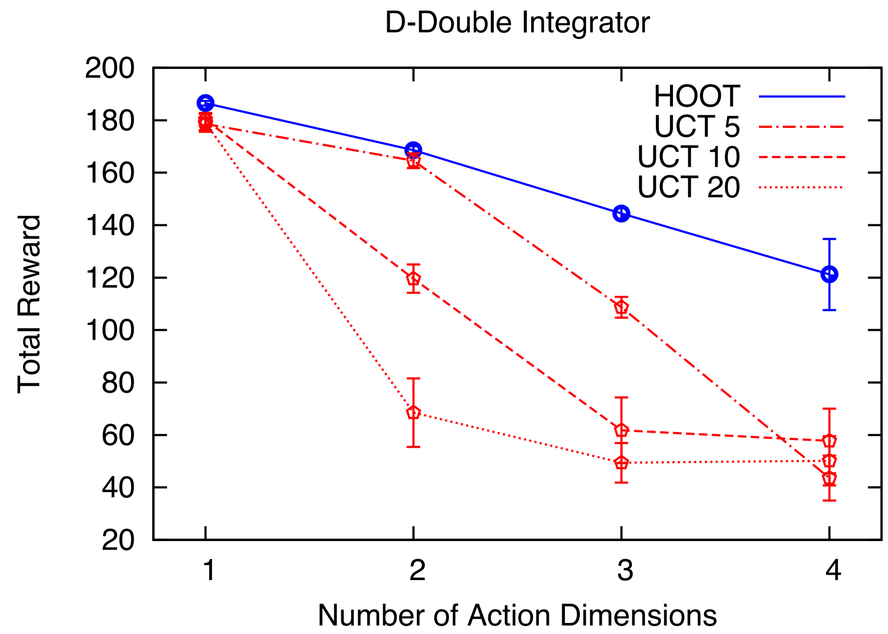
HOOT

- Our idea is to replace UCB in UCT with HOO, so that we can work directly in the continuous action space
- This leads to our algorithm HOO applied to Trees (HOOT)
- The algorithm is exactly the same as UCT, but instead of using UCB at each internal node, we maintain a HOO tree

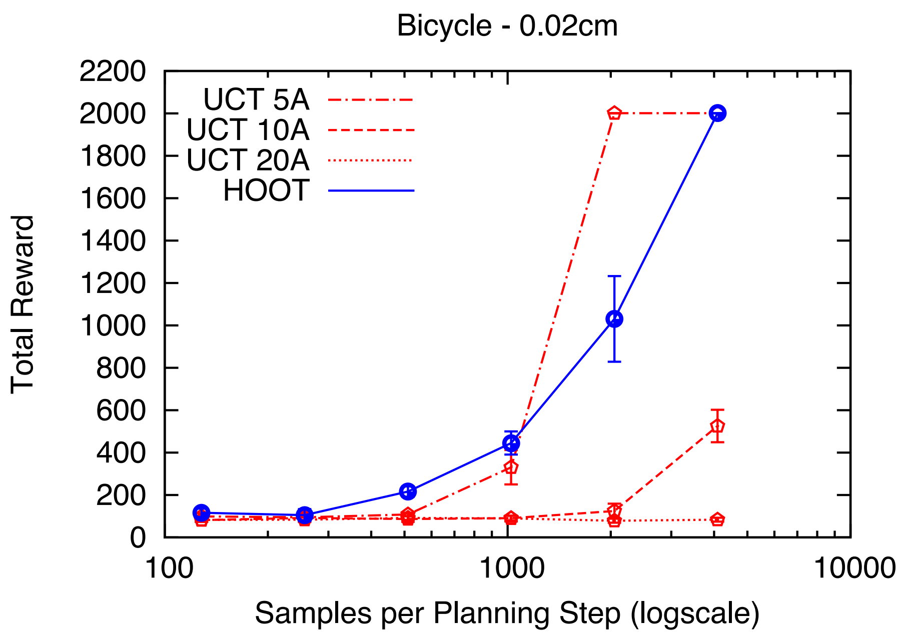
Empirical Results

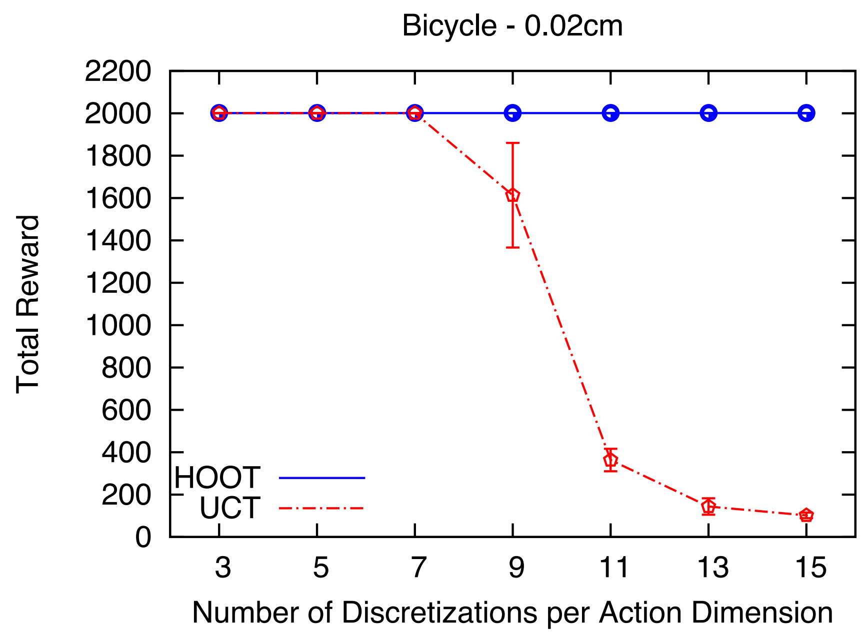






Empirical Results





Future Work

- Using HOO to optimize the n-step sequence of actions as an n-dimensional space
- Extend to continuous state spaces by a weighted interpolation between representative HOO trees

Summary

- Choosing action discretizations is non-trival!
- If you have a distance metric and your value function is locally smooth, use HOOT not vanilla UCT!

Thanks!