Kernel-Based Models for Reinforcement Learning

Nicholas K. Jong Peter Stone

Department of Computer Sciences
University of Texas at Austin

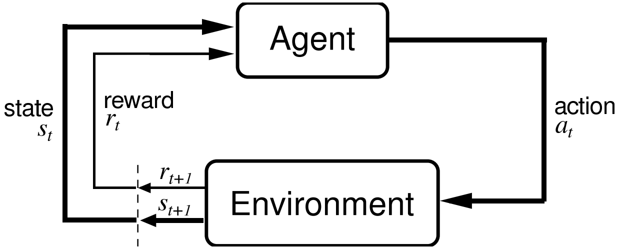
Kernel machines and Reinforcement Learning workshop,
International Conference on Machine Learning, 2006

Outline

Outline

1. Motivation
   - Model-Based Exploration
   - Function Approximation
2. Kernel-Based Approximation
   - Kernel-Based Value Functions
   - Kernel-Based Models
   - Two Kinds of Approximation
3. Empirical Results
   - Case Study
   - Benchmark Performance

Reinforcement Learning is Hard

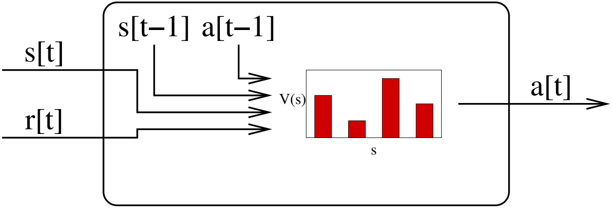


- Q-Learning made RL seem easy.
  - Convergence in the limit to optimal policy
  - Convergence for arbitrary finite Markov decision problems
- Real-world problems are too hard for current algorithms.
  - Convergence in the limit is too slow.
  - Continuous state spaces limit convergence guarantees.

Reinforcement Learning is Hard

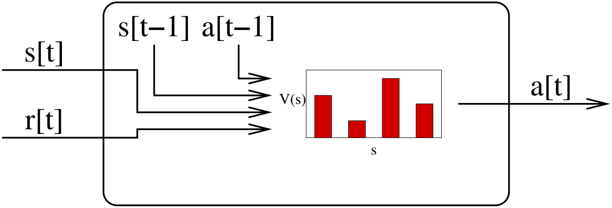
- Q-Learning made RL seem easy.
  - Convergence in the limit to optimal policy
  - Convergence for arbitrary finite Markov decision problems
- Real-world problems are too hard for current algorithms.
  - Convergence in the limit is too slow.
  - Continuous state spaces limit convergence guarantees.

Reinforcement Learning is Hard



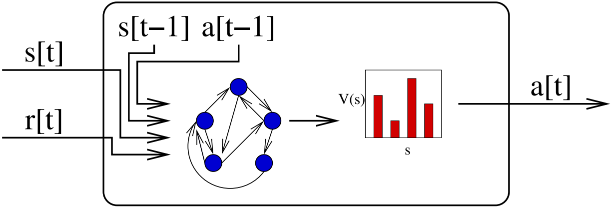
- Q-Learning made RL seem easy.
  - Convergence in the limit to optimal policy
  - Convergence for arbitrary finite Markov decision problems
- Real-world problems are too hard for current algorithms.
  - Convergence in the limit is too slow.
  - Continuous state spaces limit convergence guarantees.

Reinforcement Learning is Hard



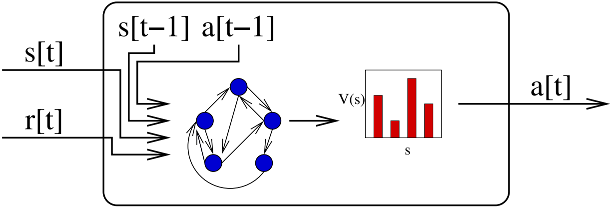
- Q-Learning made RL seem easy.
  - Convergence in the limit to optimal policy
  - Convergence for arbitrary finite Markov decision problems
- Real-world problems are too hard for current algorithms.
  - Convergence in the limit is too slow.
  - Continuous state spaces limit convergence guarantees.

Data-Efficient RL with Models



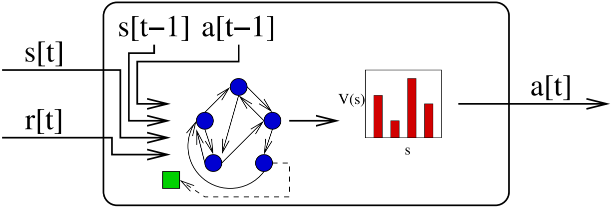
- Efficient incremental updates: Prioritized Sweeping
- More data $\implies$ accurate model
- Accurate model $\implies$ accurate value function
- Accurate value function $\implies$ good policy
- How quickly can an accurate model be learned?

Optimism in the Face of Uncertainty



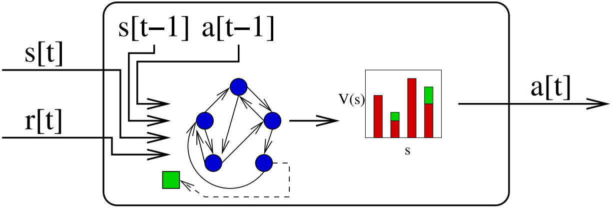
- Use model uncertainty to guide exploration.
- Assume that unfamiliar state-actions maximize value.
- Propagate optimistic values throughout value function.
- The resulting policy implicitly explores or exploits.
- This approach, from Prioritized Sweeping, underlies R-Max's polynomial sample-complexity guarantee.

Optimism in the Face of Uncertainty



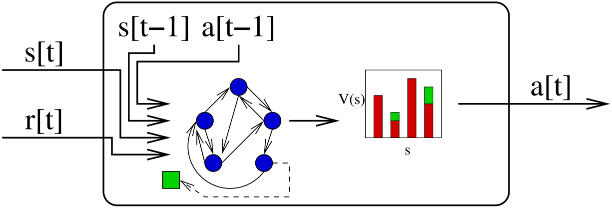
- Use model uncertainty to guide exploration.
- Assume that unfamiliar state-actions maximize value.
- Propagate optimistic values throughout value function.
- The resulting policy implicitly explores or exploits.
- This approach, from Prioritized Sweeping, underlies R-Max's polynomial sample-complexity guarantee.

Optimism in the Face of Uncertainty



- Use model uncertainty to guide exploration.
- Assume that unfamiliar state-actions maximize value.
- Propagate optimistic values throughout value function.
- The resulting policy implicitly explores or exploits.
- This approach, from Prioritized Sweeping, underlies R-Max's polynomial sample-complexity guarantee.

Optimism in the Face of Uncertainty



- Use model uncertainty to guide exploration.
- Assume that unfamiliar state-actions maximize value.
- Propagate optimistic values throughout value function.
- The resulting policy implicitly explores or exploits.
- This approach, from Prioritized Sweeping, underlies R-Max's polynomial sample-complexity guarantee.

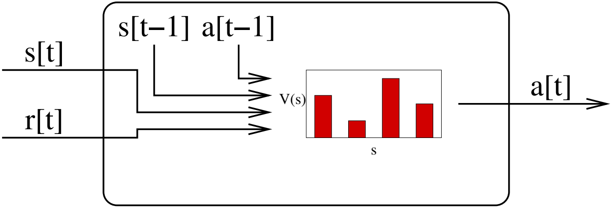
Outline

1. Motivation
   - Model-Based Exploration
   - Function Approximation

2. Kernel-Based Approximation
   - Kernel-Based Value Functions
   - Kernel-Based Models
   - Two Kinds of Approximation

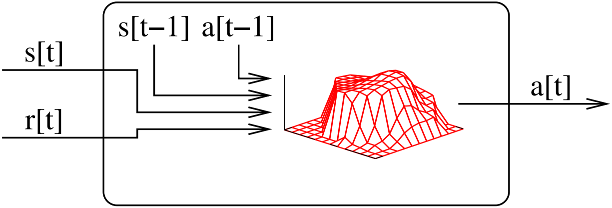
3. Empirical Results
   - Case Study
   - Benchmark Performance

Generalization to Continuous State Spaces



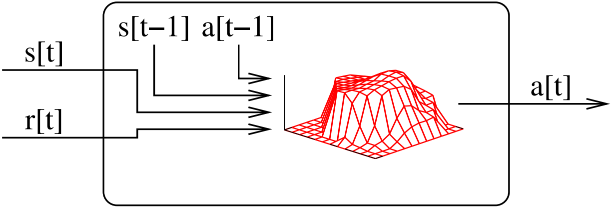
- Replace value function table with approximator.
- Limits convergence guarantees
  - Theoretical divergence in some cases
  - Only approximately optimal in most cases

Generalization to Continuous State Spaces



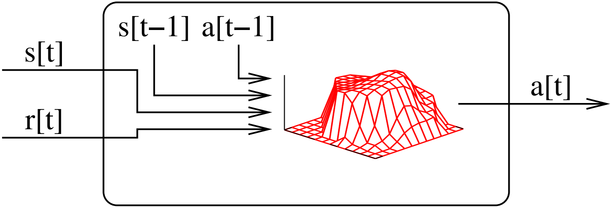
- Replace value function table with approximator.
- Limits convergence guarantees
  - Theoretical divergence in some cases
  - Only approximately optimal in most cases

Generalization to Continuous State Spaces



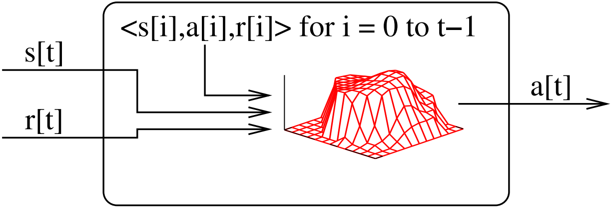
- Replace value function table with approximator.
- Limits convergence guarantees
  - Theoretical divergence in some cases
  - Only approximately optimal in most cases

Recent Trend: Offline Sample-Based Algorithms



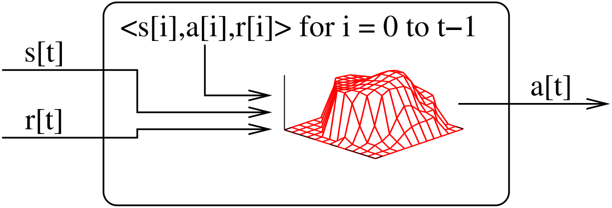
- Compute best value function from entire sample.
- Efficient use of collected data
- Facilitates theoretical analysis
  - Kernel-Based RL: convergence to optimal in the limit
- Still relies on random exploration in practice

Recent Trend: Offline Sample-Based Algorithms



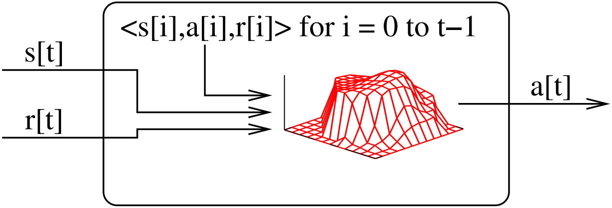
- Compute best value function from entire sample.
- Efficient use of collected data
- Facilitates theoretical analysis
  - Kernel-Based RL: convergence to optimal in the limit
- Still relies on random exploration in practice

Recent Trend: Offline Sample-Based Algorithms



- Compute best value function from entire sample.
- Efficient use of collected data
- Facilitates theoretical analysis
  - Kernel-Based RL: convergence to optimal in the limit
- Still relies on random exploration in practice

Recent Trend: Offline Sample-Based Algorithms



- Compute best value function from entire sample.
- Efficient use of collected data
- Facilitates theoretical analysis
  - Kernel-Based RL: convergence to optimal in the limit
- Still relies on random exploration in practice

Approximation and Models?

Model-free, Discrete

Q-Learning
SARSA

Model-free, Continuous

Q-Learning w/ FA
Least-Squares Policy Iteration
Kernel-Based RL

Model-based, Discrete

Prioritized Sweeping
E^3
R-Max

Model-based, Continuous

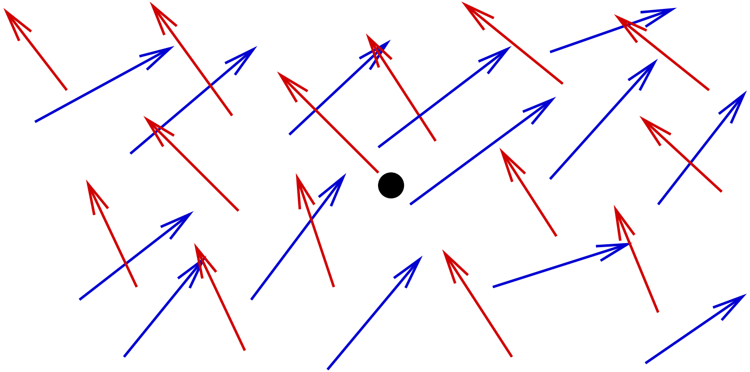
?

How to represent and reason about models of (stochastic) continuous problems?

Outline

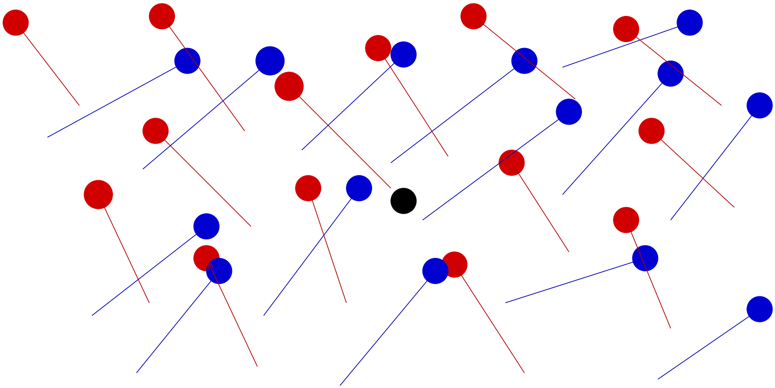
Approximating Transitions from Data

- Given: samples in the form $< s, a, r, s' >$
- Compute: $Q(s, a)$ for a given s and a



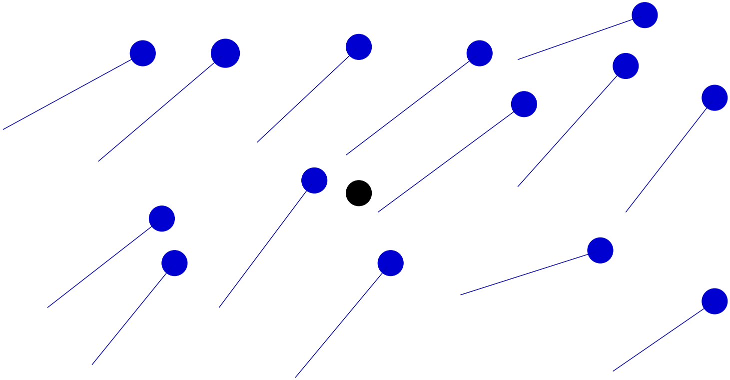
Approximating Transitions from Data

- Given: samples in the form $< s, a, r, s' >$
- Compute: $Q(s, a)$ for a given s and a



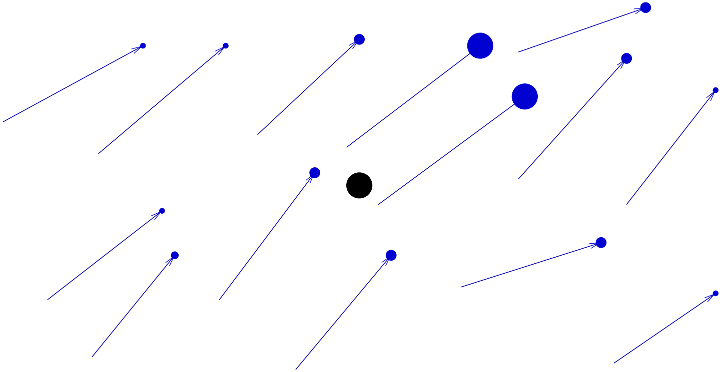
Approximating Transitions from Data

- Given: samples in the form $< s, a, r, s' >$
- Compute: $Q(s, a)$ for a given s and a



Approximating Transitions from Data

- Given: samples in the form $< s, a, r, s' >$
- Compute: $Q(s, a)$ for a given s and a



A Kernel-Based Bellman Equation

- Continuous Bellman equation:

$$Q(s,a) = R(s,a) + \gamma \int T(s,a,s')V(s')ds'$$

- A kernel-based approximation:

$$Q(s,a) = \frac{1}{Z_{s,a}} \sum_{i|a_i=a} \phi\left(\frac{d(s,s_i)}{b}\right)\left[r_i + \gamma V(s_i')\right]$$

  - d: a distance function
  - ϕ: a univariate kernel function of distance
  - b: a parameter that controls the breadth of generalization

Convergence to Optimality

- As the sample size increases, the kernel-based approximation converges in probability to the true value function if:
  - The generalization breadth b decreases at an appropriate rate.
  - An appropriate kernel (e.g. Gaussian) is used.
  - The reward function is continuous.
  - The data are uniformly sampled from the state space.
- Approximate dynamic programming for continuous problems
- Prima facie an offline algorithm

Outline

The Implicit Finite MDP

$$Q(s,a) = \frac{1}{Z_{s,a}} \sum_{i|a_i=a} \phi\left(\frac{d(s,s_i)}{b}\right)\left[r_i + \gamma V(s_i')\right]$$

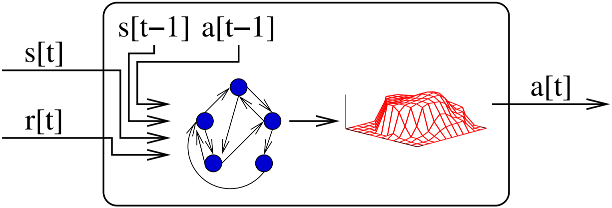
- Only finitely many states are evaluated on right-hand side.
- There exists a finite MDP for which the Bellman equations are exact.

$$\begin{aligned} T(s,a,s_i') &= \frac{1}{Z_{s,a}} \phi\left(\frac{d(s,s_i)}{b}\right), \text{if } a_i = a \\ R(s,a) &= \frac{1}{Z_{s,a}} \sum_{i|a_i=a} \phi\left(\frac{d(s,s_i)}{b}\right) r_i \end{aligned}$$

Discrete Models to Approximate Continuous Problems

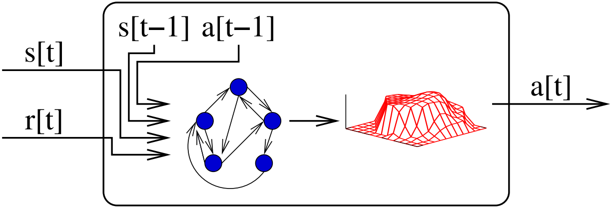
$$Q(s,a) = \frac{1}{Z_{s,a}} \sum_{i|a_i=a} \phi\left(\frac{d(s,s_i)}{b}\right)\left[r_i + \gamma V(s_i')\right]$$

- Q has a continuous domain; V has a finite domain.
- We can compute V exactly given data.
- Finite planning yields a continuous value function.



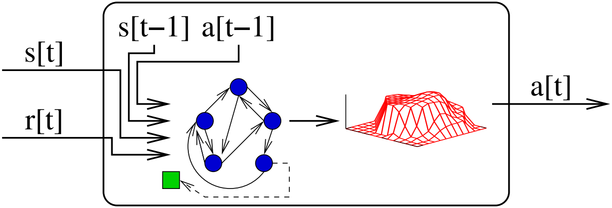
Model-Based Exploration for Continuous Problems

- Kernel-based approximation transforms continuous data into discrete data.
- We can apply model-based exploration techniques developed for finite problems.



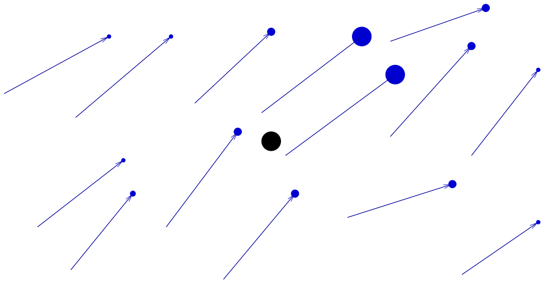
Model-Based Exploration for Continuous Problems

- Kernel-based approximation transforms continuous data into discrete data.
- We can apply model-based exploration techniques developed for finite problems.



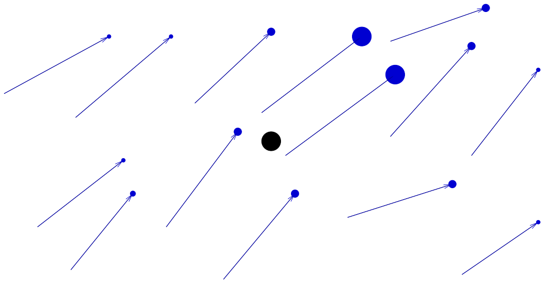
Outline

Bias Due to High Generalization



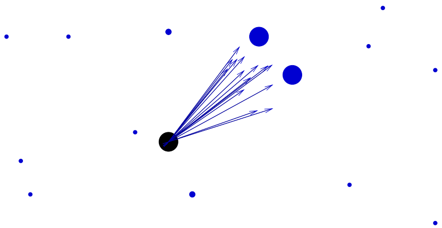
- Good empirical performance requires large generalization breadth if action effects are relative stable.
- Small generalization $\implies$ less coverage $\implies$ more data needed

Relative Transitions



- Given sample transition $s_i \rightarrow s'_i$, current state s
- Absolute transition model proposes $s' = s'_i$.
- Relative transition model proposes $s' = s + (s'_i - s_i)$.

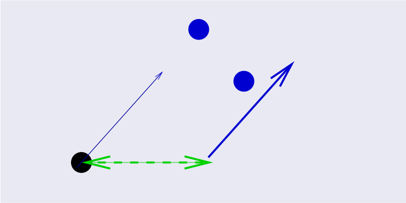
Relative Transitions



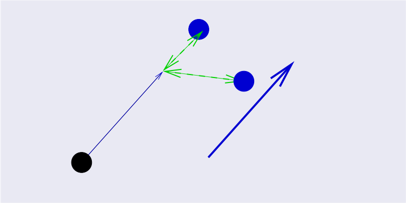
- Given sample transition $s_i \rightarrow s_i'$, current state s
- Absolute transition model proposes $s' = s_i'$.
- Relative transition model proposes $s' = s + (s_i' - s_i)$.

Approximating Transitions and Approximating Values

Approximate unknown vector with sample vectors



Approximate unknown state with sample states

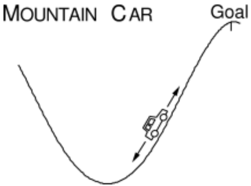


- Kernels provide weights for approximations
- Differing generalization for model and for values
- Still induces finite MDP

Outline

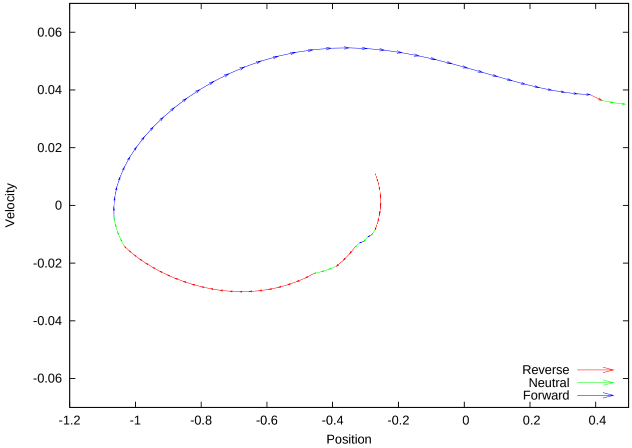
Mountain Car Domain

- Two continuous state variables
  - Horizontal position: $[-1.2, 0.5]$
  - Horizontal velocity: $[-0.07, 0.07]$
- Three actions: Reverse, Neutral, Forward
- Valley centered at position -0.5
- Underpowered motor: must go left to build kinetic energy



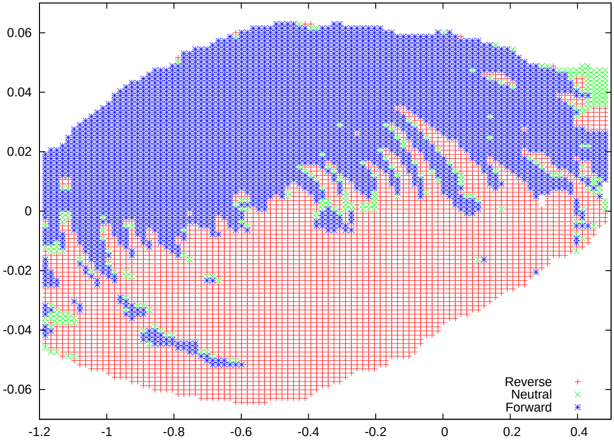
Qualitative Results

A trajectory following a learned policy:



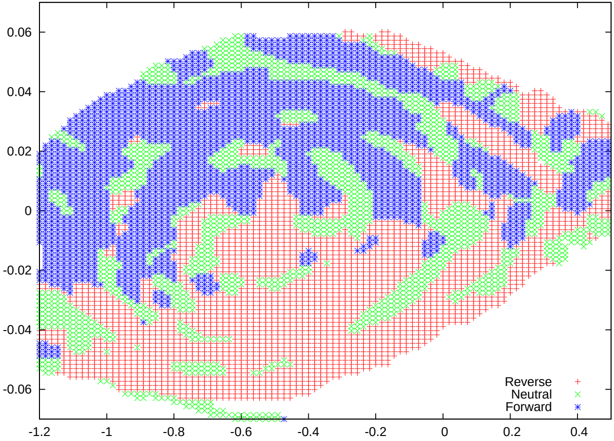
Qualitative Results

A learned policy:



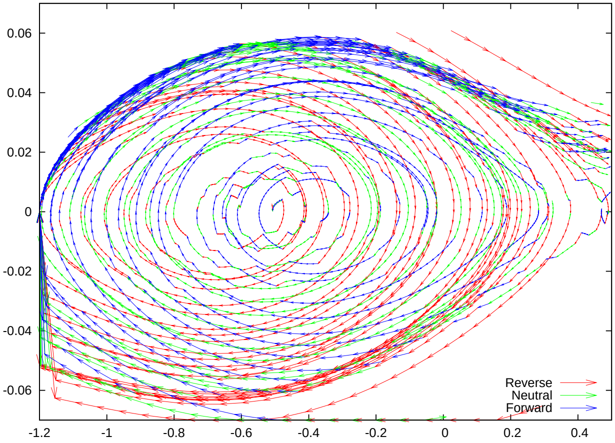
Ablation Study

A policy learned using absolute transitions:



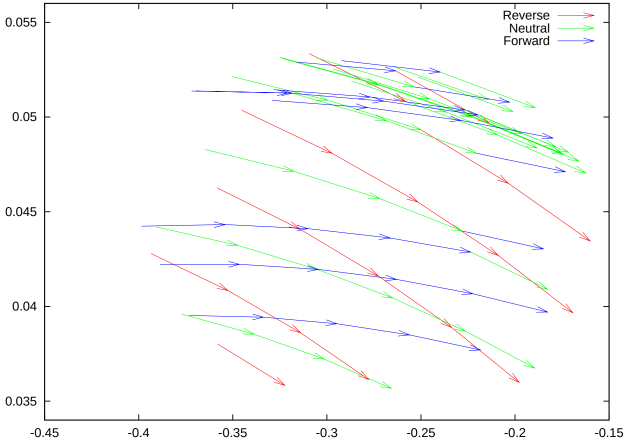
Ablation Study

A sample collected during a run using absolute transitions:



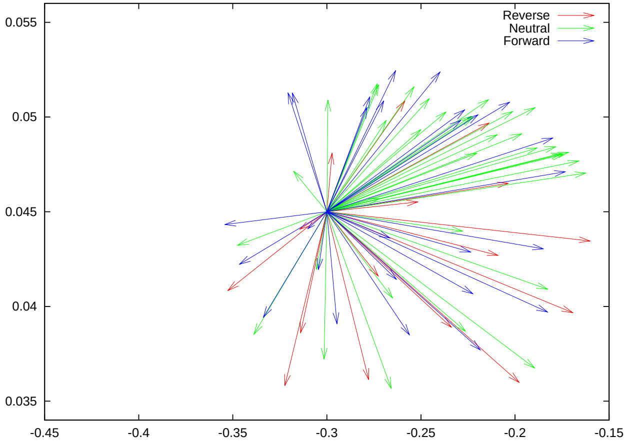
Ablation Study

A neighborhood of the data:



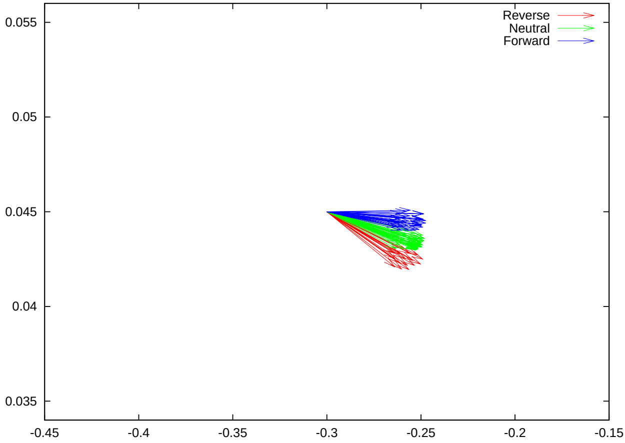
Ablation Study

Transitions predicted using absolute transitions:

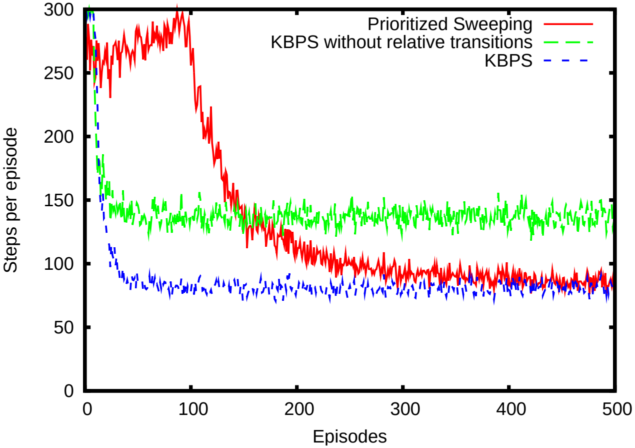


Ablation Study

Transitions predicted using relative transitions:



Ablation Study



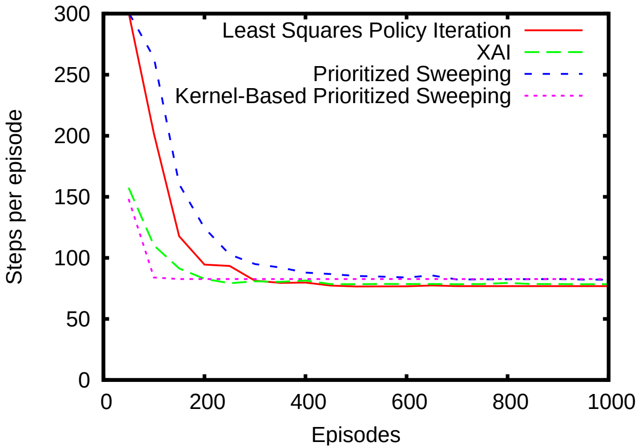
Outline

1. Motivation
   - Model-Based Exploration
   - Function Approximation
2. Kernel-Based Approximation
   - Kernel-Based Value Functions
   - Kernel-Based Models
   - Two Kinds of Approximation
3. Empirical Results
   - Case Study
   - Benchmark Performance

The NIPS 2005 RL Benchmarking Workshop

- Common interface for online RL
- Three continuous domains, including Mountain Car
- Permits comparisons against algorithms implemented and tuned by other researchers

Benchmark Results



Summary

- Approximation can be used for models instead of for value functions.
- Finite approximate models facilitate exploration in continuous problems.
- This approach yields a practical, data-efficient algorithm.

- Outlook
  - Using more sophisticated model-based exploration
  - Learning effective kernels for high-dimensional problems
  - Properties that imply convergence to optimal policies

For Further Reading I

- Atkeson, Moore, & Schaal.
  Locally weighted learning for control.
  Artificial Intelligence Review, 11:75–113, 1997.
- Moore & Atkeson.
  Prioritized sweeping: reinforcement learning with less data and less real time.
  Machine Learning, 13:103–130, 1993.
- Ormoneit & Sen.
  Kernel-based reinforcement learning.
  Machine Learning, 49(2):161–178, 2002.