Bayesian Sparse Sampling for On-line Reward Optimization

Tao Wang
Daniel Lizotte
Michael Bowling
Dale Schuurmans
Background Perspective

- Be Bayesian about reinforcement learning
- Ideal representation of uncertainty for action selection

Why are Bayesian approaches not prevalent in RL?

- Attack computational barriers
Main Contribution

- Sparser *sparse sampling* for Bayesian RL
- Controllable computational cost
- Higher quality action selection than current methods
  - Greedy
  - Epsilon - greedy
  - Boltzmann  *(Luce 1959)*
  - Thompson Sampling  *(Thompson 1933)*
  - Bayes optimal  *(Hee 1978)*
  - Interval estimation  *(Lai 1987, Kaelbling 1994)*
  - Myopic value of perfect info.  *(Dearden, Friedman, Andre 1999)*
  - Standard sparse sampling  *(Kearns, Mansour, Ng 2001)*
  - Péret & Garcia  *(Péret & Garcia 2004)*

- (Almost) completely general
Sequential Decision Making

How to make an optimal decision?

"Planning"

Requires model $P(r,s'|s,a)$

General case: Fixed point equations:

$$V(s) = \sup_a Q(s,a)$$

$$Q(s,a) = E_{r,s'|s,a} [r + \gamma V(s')]$$

This is: finite horizon, finite action, finite reward case
Reinforcement Learning

Do not have model $P(r,s'|s,a)$
Reinforcement Learning

Cannot Compute $E_{r,s'|s,a}$

Do not have model $P(r,s'|s,a)$
Reinforcement Learning

Standard approach: keep point estimate
e.g. via local Q-value estimates

Problem: greedy does not explore

[Diagram showing reinforcement learning structure with decision points and outcomes]

Do not have model P(r,s'|s,a)

How to select action?

Greedy
Reinforcement Learning

Problem: do not account for uncertainty in estimates

How to explore?

ε -greedy

Boltzmann
Reinforcement Learning

Intuition:
greater uncertainty  →  greater potential

Problem:  δ ’s computed myopically: doesn’t consider horizon
Bayesian Reinforcement Learning

Prior $P(\theta)$ on model $P(rs'|sa, \theta)$  
Belief state $b = P(\theta)$

**meta-level state**

Choose action to maximize long term reward

Meta-level MDP

<table>
<thead>
<tr>
<th>Decision</th>
<th>Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>$r, s', b'$</td>
</tr>
</tbody>
</table>

Meta-level Model

$P(r, s'b'|s, b, a)$

<table>
<thead>
<tr>
<th>Decision</th>
<th>Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a'$</td>
<td>$r', s'', b''$</td>
</tr>
</tbody>
</table>

Meta-level Model

$P(r', s''b''|s', b', a')$

Have a model for meta-level transitions!

- based on posterior update and expectations over base-level MDPs
Bayesian Reinforcement Learning

How to make an optimal decision?

Solve planning problem in meta-level MDP:
- Optimal Q, V values

Problem: meta-level MDP much larger than base-level MDP
Impractical
Bayesian RL Decision Making

Current approximation strategies:

Consider current belief state \( b \)

Draw a base-level MDP

Greedy approach:

\[ \text{current } b \rightarrow \text{mean base-level MDP model} \]
\[ \rightarrow \text{point estimate for } Q, V \]
\[ \rightarrow \text{choose greedy action} \]

But doesn’t consider uncertainty
Bayesian RL Decision Making

Current approximation strategies:

Consider current belief state $b$

Draw a base-level MDP

Thompson approach:

current $b$ $\to$ sample a base-level MDP model
$\to$ point estimate for $Q$, $V$

(Choose action proportional to probability it is max $Q$)

Exploration is based on uncertainty
But still myopic
Contribution

• Try to better approximate Bayes optimal action selection by performing lookahead

• Adapt “sparse sampling” (Kearns, Mansour, Ng)
  – Make some practical improvements
Sparse Sampling

(Kearns, Mansour, Ng 2001)

Approximate values
Enumerate action choices
Subsample action outcomes
Bound depth
Back up approx values

+ Chooses approximately optimal action with high probability
  (if depth, sampling large enough)

- Achieving guarantees too expensive

+ But can control depth, sampling
Bayesian Sparse Sampling
Bayesian Sparse Sampling
Observation 1

• Do not need to enumerate actions in a Bayesian setting
  – Given random variables $Q_1, \ldots, Q_K$
  – and a prior $P(Q_1, \ldots, Q_K)$
  – Can approximate $\max(Q_1, \ldots, Q_K)$
  – Without observing every variable

(Stop when posterior probability of a significantly better Q-value is small)
Bayesian Sparse Sampling
Observation 2

- Action value estimates are not equally important
  - Need better Q value estimates for some actions but not all
  - Preferentially expand tree under actions that might be optimal

Biased tree growth
Use Thompson sampling to select actions to expand
Bayesian Sparse Sampling
Observation 3

Correct leaf value estimates to same depth

Use mean MDP Q-value multiplied by remaining depth

effective horizon N=3
Bayesian Sparse Sampling
Observation 4

Include greedy action at decision nodes (if not sampled)

Add greedy action for local belief state
Bayesian Sparse Sampling

Tree growing procedure

1. Sample prior for a model
2. Solve action values
3. Select the optimal action

• Descend sparse tree from root
  – Thompson sample actions
  – Sample outcome

• Until new node added
• Repeat until tree size limit reached

Control computation by controlling tree size
Simple experiments

- 5 Bernoulli bandits $a_1, \ldots, a_5$
- Beta priors
- Sampled model from prior
- Run action selection strategies
- Repeat 3000 times
- Average accumulated reward per step
Five Bernoulli Bandits

Average Reward per Step vs Horizon

- eps-Greedy
- Boltzmann
- Interval Est.
- Thompson
- MVPI
Five Bernoulli Bandits

Average Reward per Step vs Horizon

- eps-Greedy
- Boltzmann
- Interval Est.
- Thompson
- MVPI
- Sparse Samp.
- Bayes Samp.
Simple experiments

- 5 Gaussian bandits $a_1, \ldots, a_5$
- Gaussian priors
- Sampled model from prior
- Run action selection strategies
- Repeat 3000 times
- Average accumulated reward per step
Gaussian process bandits

- General action spaces
  - Continuous actions, multidimensional actions
- Gaussian process prior over reward models
  - Covariance kernel between actions
- Action rewards correlated
- Posterior is a Gaussian process
Gaussian process experiments

- 1 dimensional continuous action space
- GP priors RBF kernel
- Sampled model from prior
- Run action selection strategies
- Repeat 3000 times
- Average accumulated reward per step
1-dimensional Continuous Gaussian Process

Average Reward per Step vs. Horizon

- eps-Greedy
- Boltzmann
- Interval Est.
- Thompson
- MVPI
1-dimensional Continuous Gaussian Process

Horizon

Average Reward per Step

eps-Greedy
Boltzmann
Interval Est.
Thompson
MVPI
Sparse Samp.
Bayes Samp.

ICML 05
Gaussian process experiments

- 2 dimensional continuous action space
- GP priors RBF kernel
- Sampled model from prior
- Run action selection strategies
- Repeat 3000 times
- Average accumulated reward per step
2-dimensional Continuous Gaussian Process

Average Reward per Step vs Horizon

- eps-Greedy
- Boltzmann
- Interval Est.
- Thompson
- MVPI
2-dimensional Continuous Gaussian Process

Average Reward per Step vs Horizon

- eps-Greedy
- Boltzmann
- Interval Est.
- Thompson
- MVPI
- Sparse Samp.
- Bayes Samp.
Gaussian Process Bandits

• Very flexible model
• Actions can be complicated
  – e.g. a parameterized policy
  – Just need a kernel between policies
• Applications in robotics & game playing
• Reward = total reward accumulated by a policy in an episode
Summary

Bayesian sparse sampling

• Flexible and practical technique for improving action selection

• Reasonably straightforward

• Bandit problems
  – Planning is “easy”
    (at least approximate planning is “easy”)
Future Work

- Built code base for complicated domains
- Conducted proof-of-concept testing on robot (PIONEER 3-DX)

- To analyze the complexity of algorithms theoretically
- To develop Applications
  - Kuhn poker (game)
  - AIBO walking (robotics)
  - VendorBot (robotics)
- To design better strategies for action selection
Thank You!