Gradient Reinforcement Learning of POMDP Policy Graphs

Douglas Aberdeen

Research School of Information Science and Engineering

Australian National University

Jonathan Baxter

WhizBang! Labs

July 23, 2001

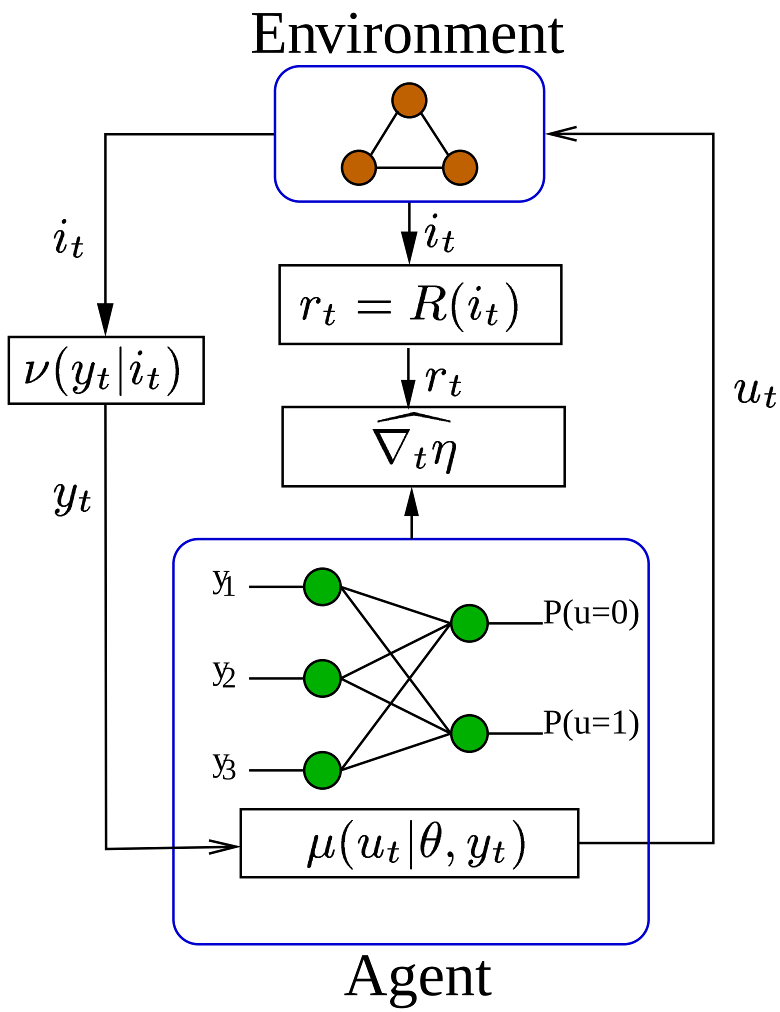
Outline

- *Motivation*
- GPOMDP, a policy gradient RL algorithm
- GPOMDP with I-state
- The Load-Unload problem
- Related Work
- Pros and Cons of GPOMDP with I-state
- Repairing I-state GPOMDP
- The Heaven-Hell problem
- (?) Using prior knowledge to reduce gradient variance

Goals

- Develop scalable RL algorithms that learn near optimal controls for POMDPs without prior knowledge of the model. This is *hard!*
- Demonstrate these algorithms on a large scale, real world problems:
  - speech processing;
  - robot navigation.

A POMDP



Environment
i_t
i_t
$r_t = R(i_t)$
r_t
$\nu(y_t|i_t)$
$\widehat{\nabla_t \eta}$
u_t
y_t
y_1
y_2
y_3
P(u=0)
P(u=1)
$\mu(u_t|\theta, y_t)$
Agent

Historical Perspective I

Bellman's Equation

Richard Bellman (1957)

$$\mathbf{J}^* = \mathbf{r} + \beta \mathbf{P} \mathbf{J}^*.$$

- Describes n_s equations with n_s unknowns (n_s = states).
- Model must be known.
- This formulation is for MDPs only.
- Intractable for more than a few tens of states.

Historical Perspective II

Policy Iteration

Bellman (1957) and Howard (1960)

- Finds a solution to the Bellman equation via dynamic programming.
- Practical for much larger state spaces.
- Related method: value iteration.
- Function approximation for RL in use by 1965 (Waltz and Fu 1965).

Historical Perspective III

Simulated Methods

- Do not require the environment model. They learn from experience.
- `Q-learning` (Watkin's 1989).
- Eligibility traces: $\texttt{TD}(\lambda)$ (Sutton 1988).

Historical Perspective IV

Policy Gradient Methods

- Learns the policy directly.
- Nice convergence properties, even for function approximators.
- Variance in the gradient estimates is a problem.
- `REINFORCE` (Williams 1992).
- `GPOMDP` (Baxter & Bartlett 1999).
- Hybrids: `VAPS` (Baird & Moore 1999).

Historical Perspective V

Exact POMDP methods

Aström (1965), Sondik (1971)

- Re-introduces the environment model.
- Modified Bellman equation computes the value of *belief* states.
- At least PSpace-complete so approximate methods are needed.

Controlling POMDPs sans model, with infinite state and action spaces, is about as general as it gets.

Failings of current methods

The drawbacks of current approximate POMDP methods include:

- Assumption of a model of the environment.
- Only recalling events finitely far into the past.
- Use of an independent internal state model that does not aim to maximise the long term reward.
- Do not easily generalize to continuous observations and actions.
- Applications to toy problems only.

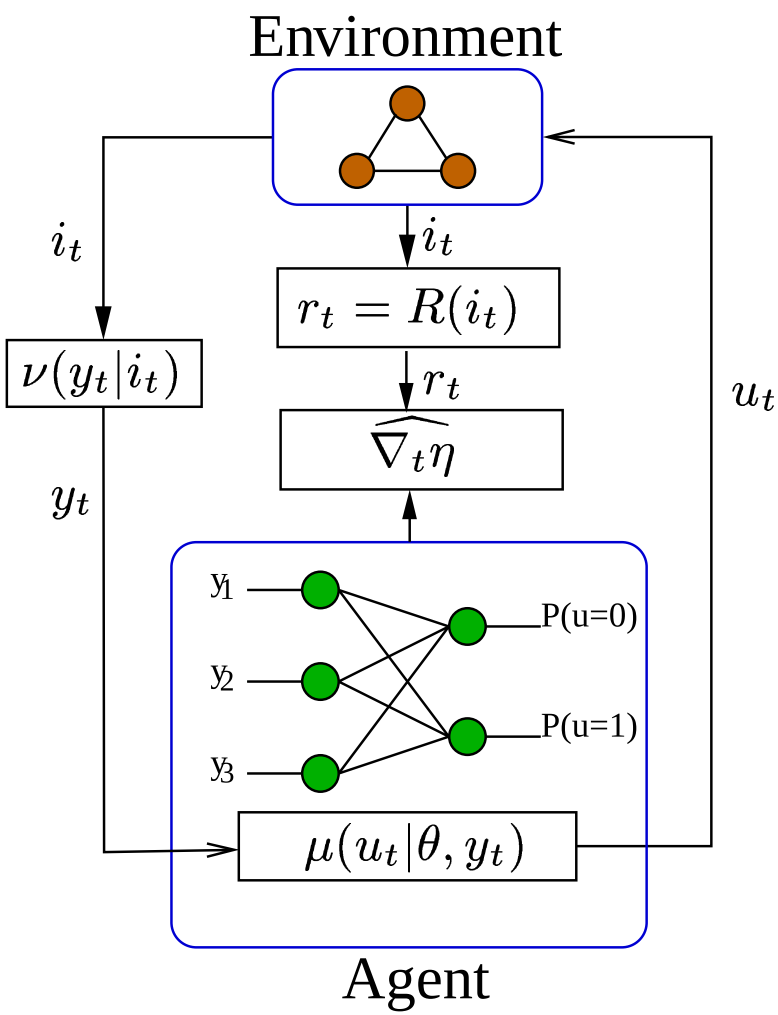
Outline

- Motivation
- *GPOMDP, a policy gradient RL algorithm*
- GPOMDP with I-state
- The Load-Unload problem
- Related Work
- Pros and Cons of GPOMDP with I-state
- Repairing I-state GPOMDP
- The Heaven-Hell problem
- (?) Using prior knowledge to reduce gradient variance

The GPOMDP algorithm

GPOMDP is a policy gradient approach to reinforcement learning.

- GPOMDP is an algorithm for of estimating the gradient of $\eta = \lim_{T \to \infty} \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} r_t\right]$ with respect to the parameters of the policy.
- Estimates the infinite horizon average reward gradient by using a parameter β which is equivalent to discounting.
- Computes $\frac{1}{T}\sum_{t=0}^{T-1} \frac{\nabla\mu(u_t|\theta,y_t)}{\mu(u_t|\theta,y_t)} \sum_{s=t+1}^{T} \beta^{s-t-1} r_s$.
- Works for POMDP environments if observations are belief states.
- Similar to REINFORCE (Williams 1992) and VAPS (Baird & Moore 1999), (Marbach & Tsitsiklis 1999).

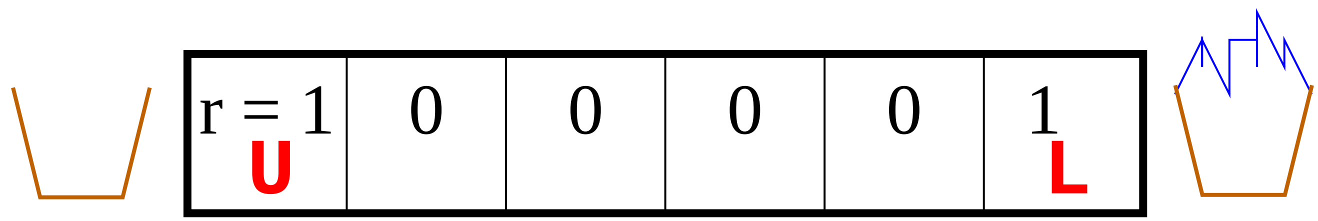

Environment
i_t
i_t
$r_t = R(i_t)$
r_t
$\nu(y_t|i_t)$
$\widehat{\nabla_t \eta}$
u_t
y_t
y1
y2
y3
P(u=0)
P(u=1)
$\mu(u_t|\theta, y_t)$
Agent

Outline

- Motivation
- GPOMDP, a policy gradient RL algorithm
- *GPOMDP with I-state*
- The Load-Unload problem
- Related Work
- Pros and Cons of GPOMDP with I-state
- Repairing I-state GPOMDP
- The Heaven-Hell problem
- (?) Using prior knowledge to reduce gradient variance

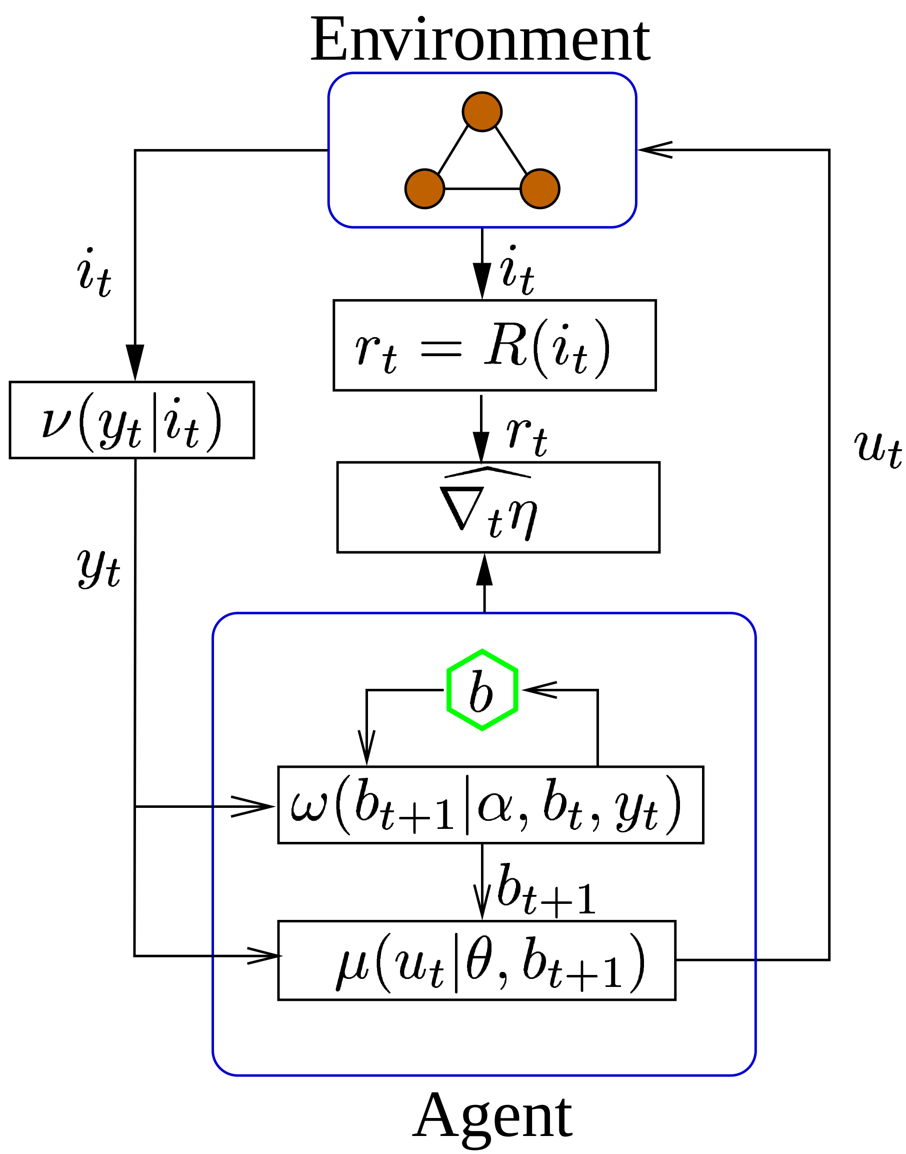
GPOMDP with memory

- GPOMDP implements a memoryless controller, which is not always sufficient



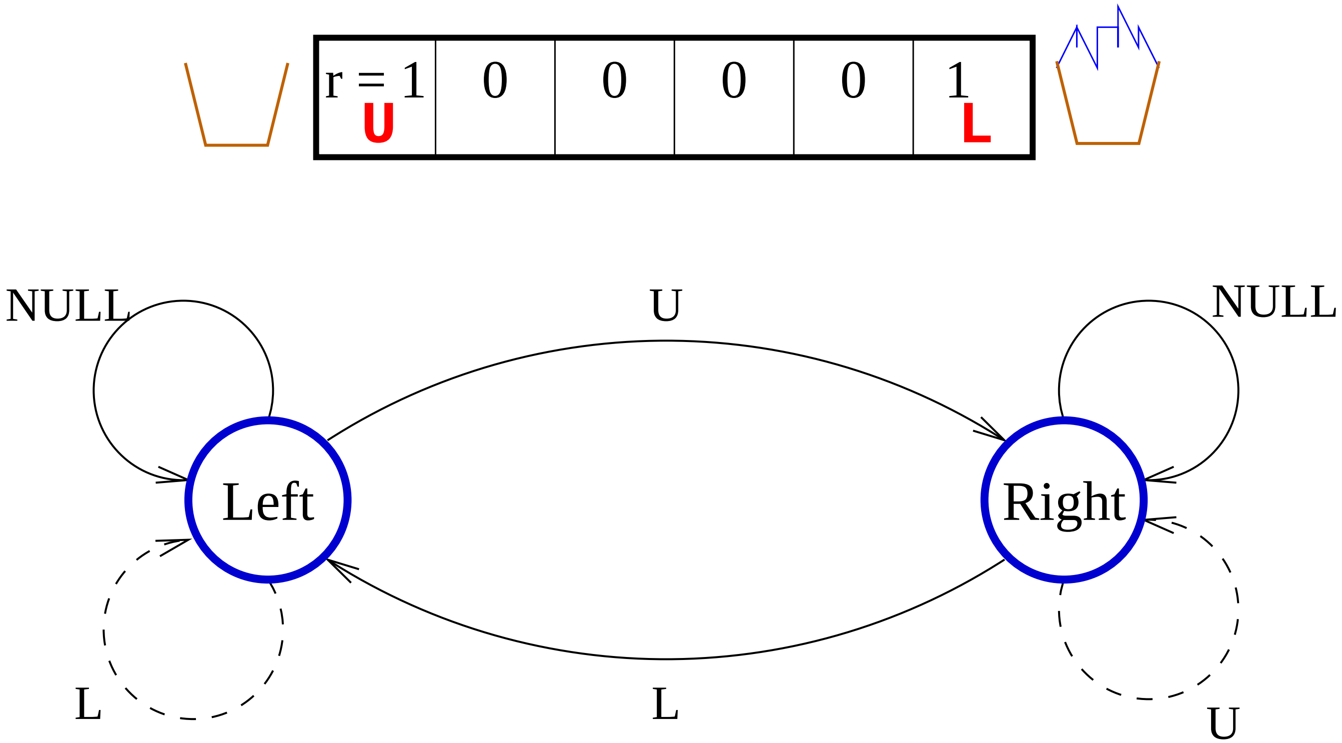
(Peshkin, Meuleau, Kaebling 1999)

- GPOMDP has been extended with *I-states* $b_t \in \mathcal{B}$.
- $\omega(b_{t+1}|\alpha, b_t, y_t)$ gives the next I-state probabilities.
- $\mu(u_t|\theta, b_{t+1})$ gives action probabilities.
- GPOMDP computes the gradient w.r.t θ and α independently.

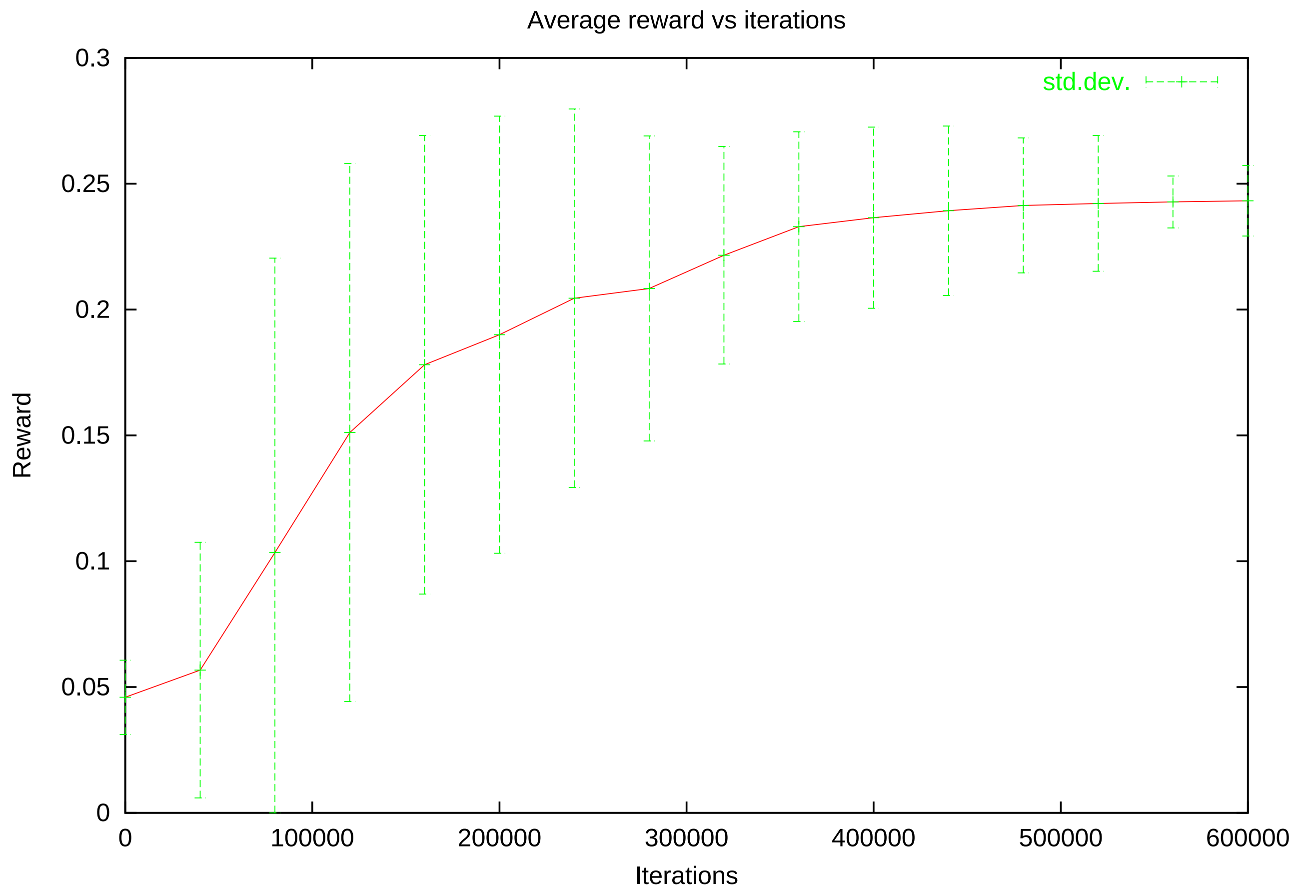

Environment
i_t
i_t
$r_t = R(i_t)$
$\nu(y_t|i_t)$
r_t
u_t
$\widehat{\nabla_t \eta}$
y_t
b
$\omega(b_{t+1}|\alpha, b_t, y_t)$
b_{t+1}
$\mu(u_t|\theta, b_{t+1})$
Agent

Outline

- Motivation
- GPOMDP, a policy gradient RL algorithm
- GPOMDP with I-state
- *The Load-Unload problem*
- Related Work
- Pros and Cons of GPOMDP with I-state
- Repairing I-state GPOMDP
- The Heaven-Hell problem
- (?) Using prior knowledge to reduce gradient variance



Policy graph learnt for the Load/Unload problem.



Convergence of the Load/Unload problem using 4 I-States.

(Averaged over 100 runs.)

Outline

- Motivation
- GPOMDP, a policy gradient RL algorithm
- GPOMDP with I-state
- The Load-Unload problem
- *Related Work*
- Pros and Cons of GPOMDP with I-state
- Repairing I-state GPOMDP
- The Heaven-Hell problem
- (?) Using prior knowledge to reduce gradient variance

Related Work

- Use HMMs to learn the model (Chrisman 1992).
- Recurrent Neural Networks (Lin & Mitchell 1992).
- Differentiable approx. to piecewise function (Parr & Russell 1995).
- `U-Tree`'s: Dynamic finite history windows (McCallum 1996).
- External memory setting actions (Peshkin, Meuleau, Kaebling 1999).

Pros of GPOMDP with I-states

- Converges to the optimal policy that can be learnt with n_b I-states.
- Does not require a model of the POMDP.
- I-states can remember occurrences infinitely far into the past.
- Works with continuous state and action spaces.
- Theoretically scales to large problems.

Cons of GPOMDP with I-states

1. GPOMDP has a large variance as $\beta \to 1$.
2. I-states increase the mixing time of the overall system.
   - Importance Sampling;
   - replace μ with an MDP alg. that works on the I-states;
   - eligibility trace filtering to incorporate prior knowledge;
   - deterministic $\mu(u_t|b_{t+1}, y_t, a_t)$.
3. Internal states are initially undifferentiated, resulting in $\nabla\eta \approxeq 0$.
   - Define a sparse internal finite state machine.

Outline

- Motivation
- GPOMDP, a policy gradient RL algorithm
- GPOMDP with I-state
- The Load-Unload problem
- Related Work
- Pros and Cons of GPOMDP with I-state
- *Repairing I-state GPOMDP*
- The Heaven-Hell problem
- (?) Using prior knowledge to reduce gradient variance

Undifferentiated I-states I

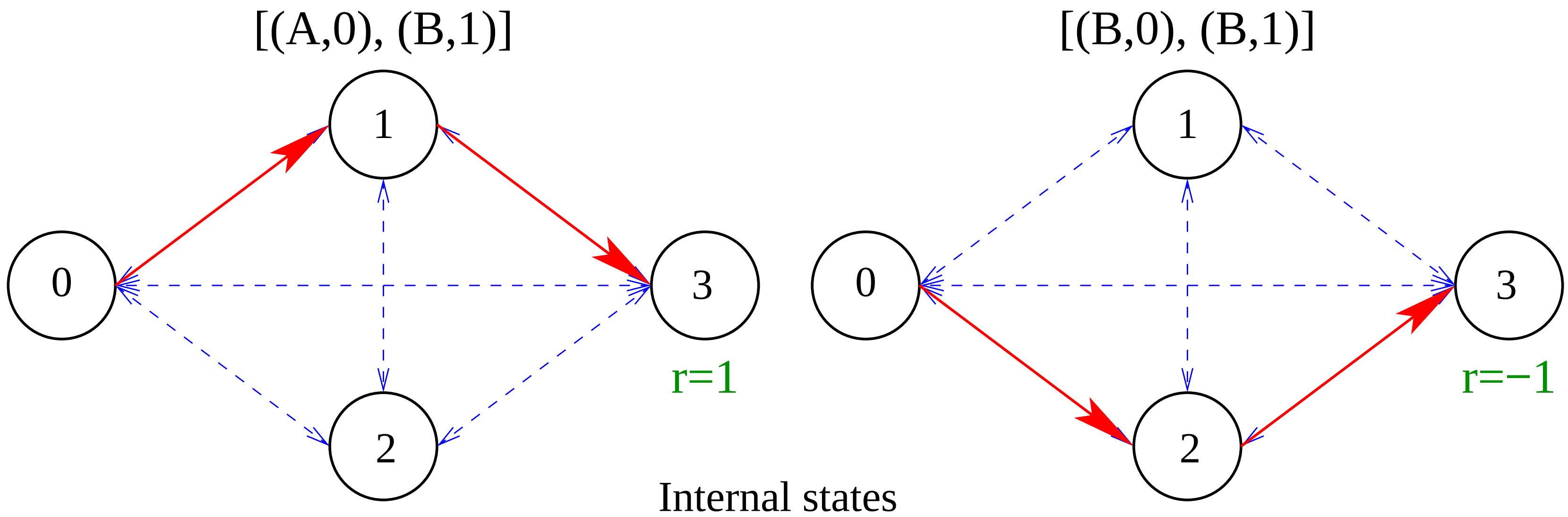


Figure 1: Possible I-state trajectories for observation/action trajectories.

Undifferentiated I-states II

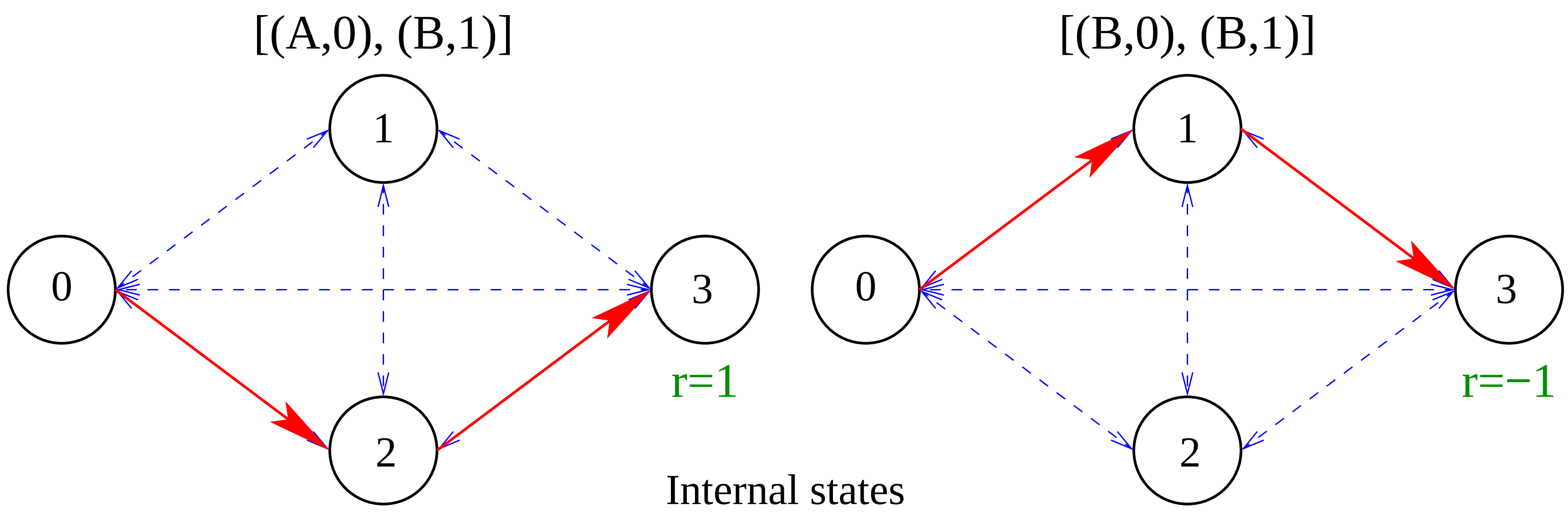


Figure 2: Alternate, equally likely, I-state trajectories.

Sparse transitions for I-states

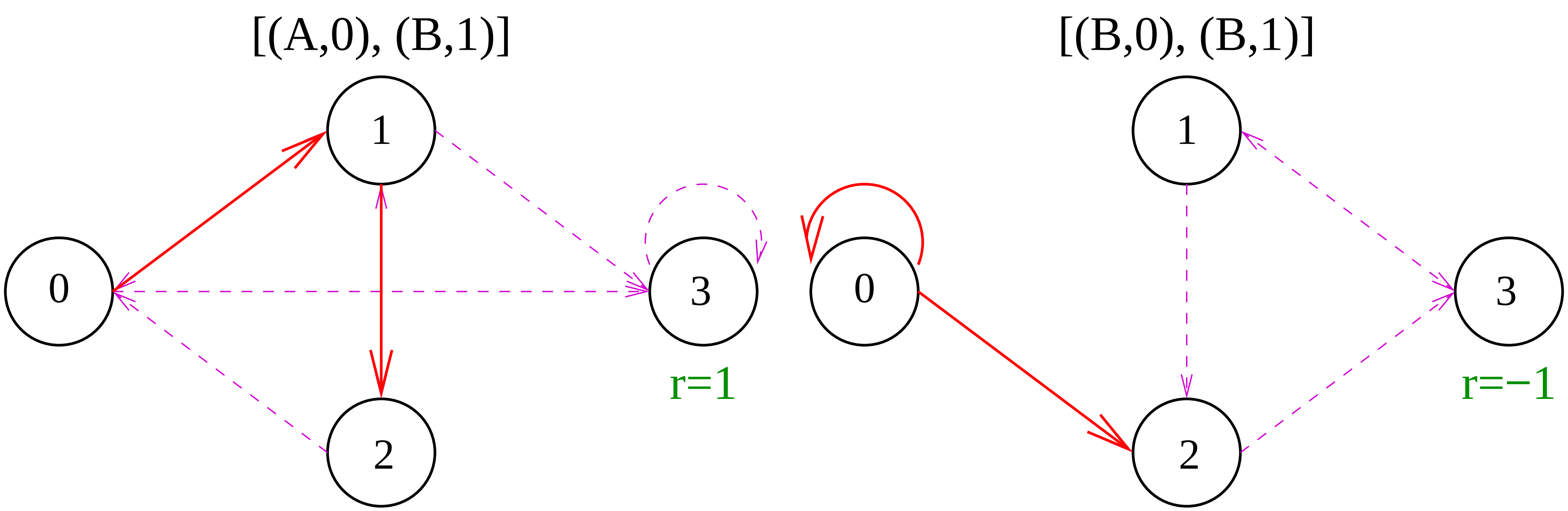


Figure 3: Reduced number of possible I-state trajectories.

Signpost problem description

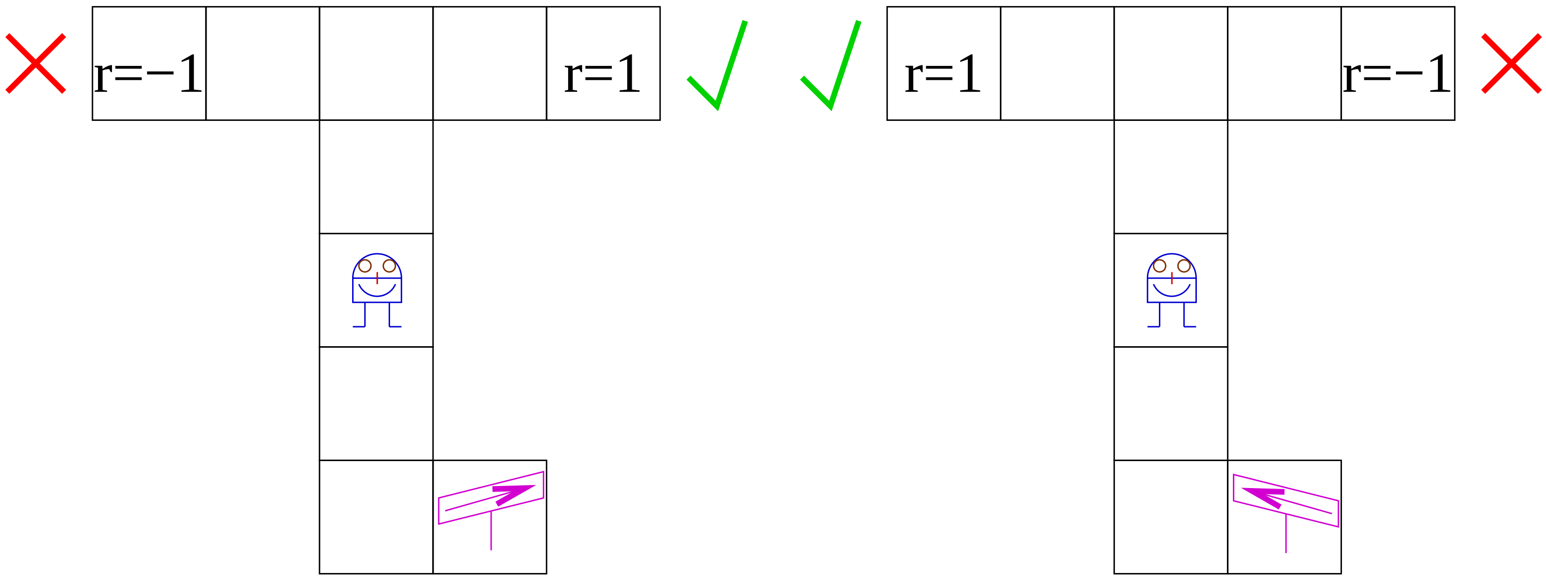
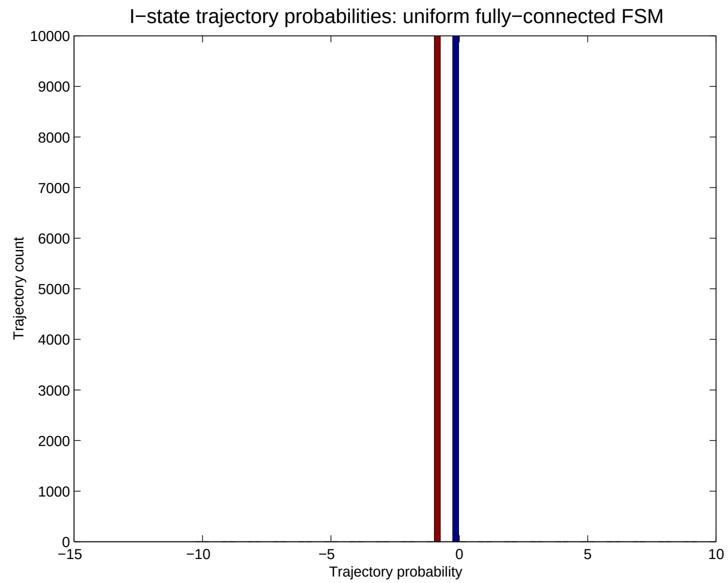
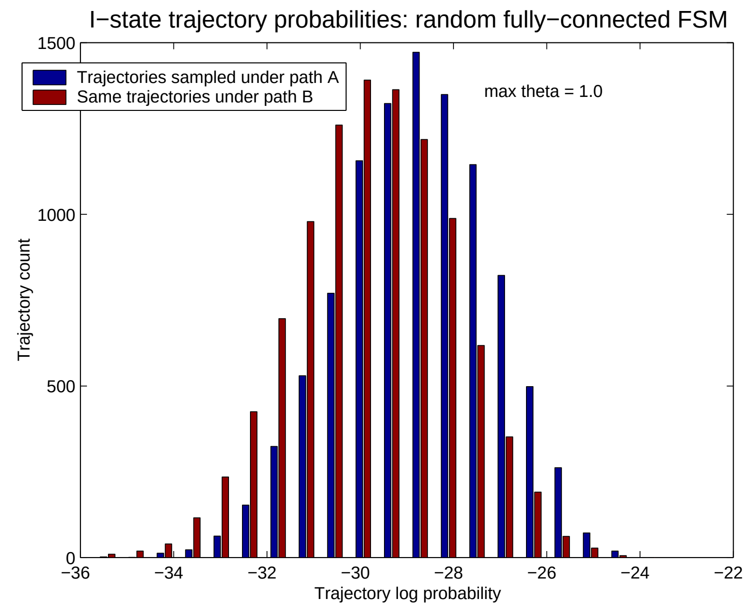


Figure 4: Simplified Heaven-Hell problem. Agent must visit lower state to determine which way to move at the top of the T (Thrun 2000), (Geffner & Bonet 1998).

I-states Trajectory Probabilities



(a)



(b)

Figure 5: Histogram of probabilities of 10,000 I-state trajectories sampled from the signpost problem. Shown for 2 sets of observations.

Sparse transitions for I-states

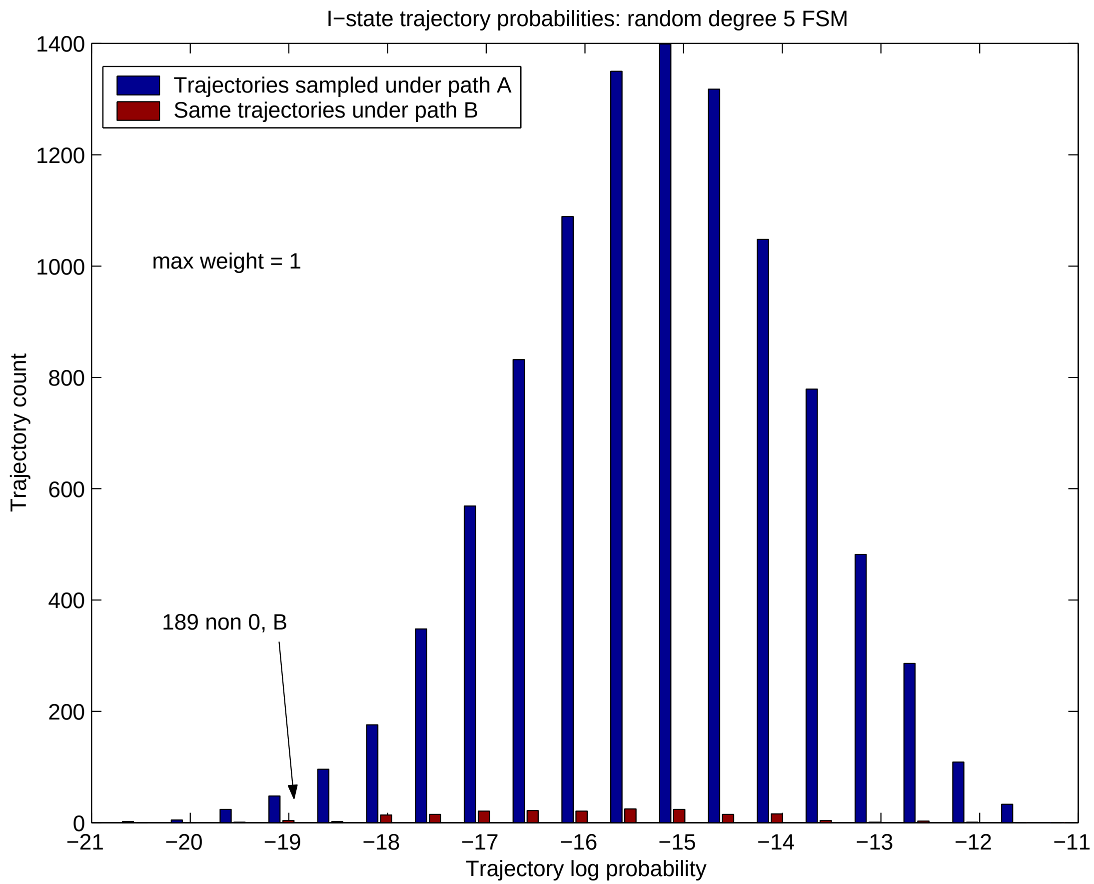


Figure 6: I-state trajectory histograms for sparse I-state transitions.

Outline

- Motivation
- GPOMDP, a policy gradient RL algorithm
- GPOMDP with I-state
- The Load-Unload problem
- Related Work
- Pros and Cons of GPOMDP with I-state
- Repairing I-state GPOMDP
- *Heaven-Hell problem*
- (?) Using prior knowledge to reduce gradient variance

Outline

- Motivation
- GPOMDP, a policy gradient RL algorithm
- GPOMDP with I-state
- The Load-Unload problem
- Related Work
- Pros and Cons of GPOMDP with I-state
- Repairing I-state GPOMDP
- Heaven-Hell problem
- *Using prior knowledge to reduce gradient variance*

A simple POMDP?

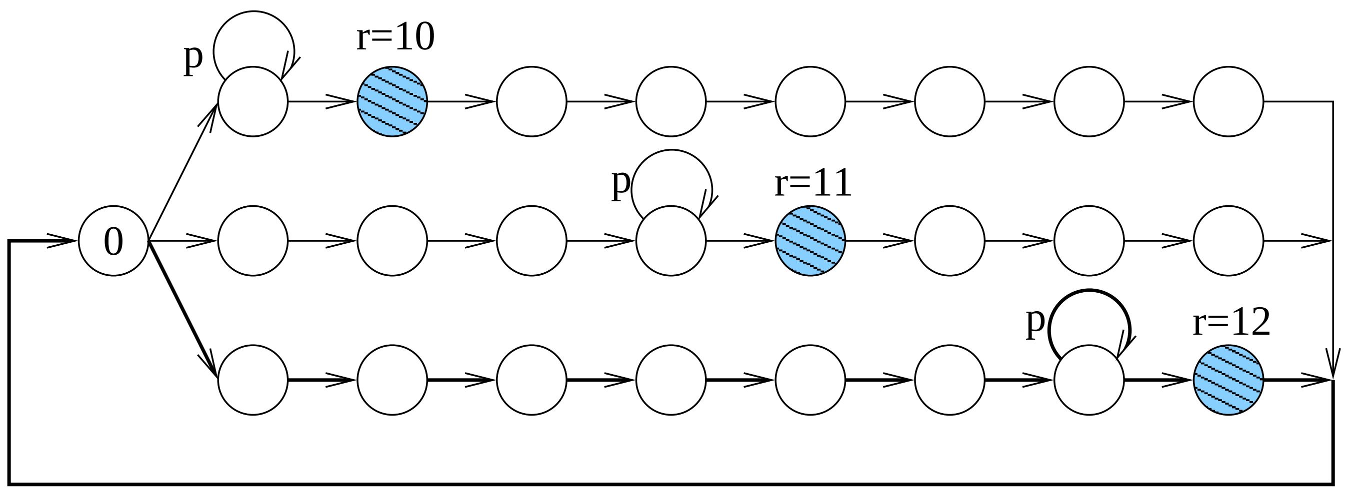


Figure 7: A UMDP which requires $\beta > 0.97$ for `GPOMDP` to learn to act optimally.

GPOMDP Eligibility Trace Update

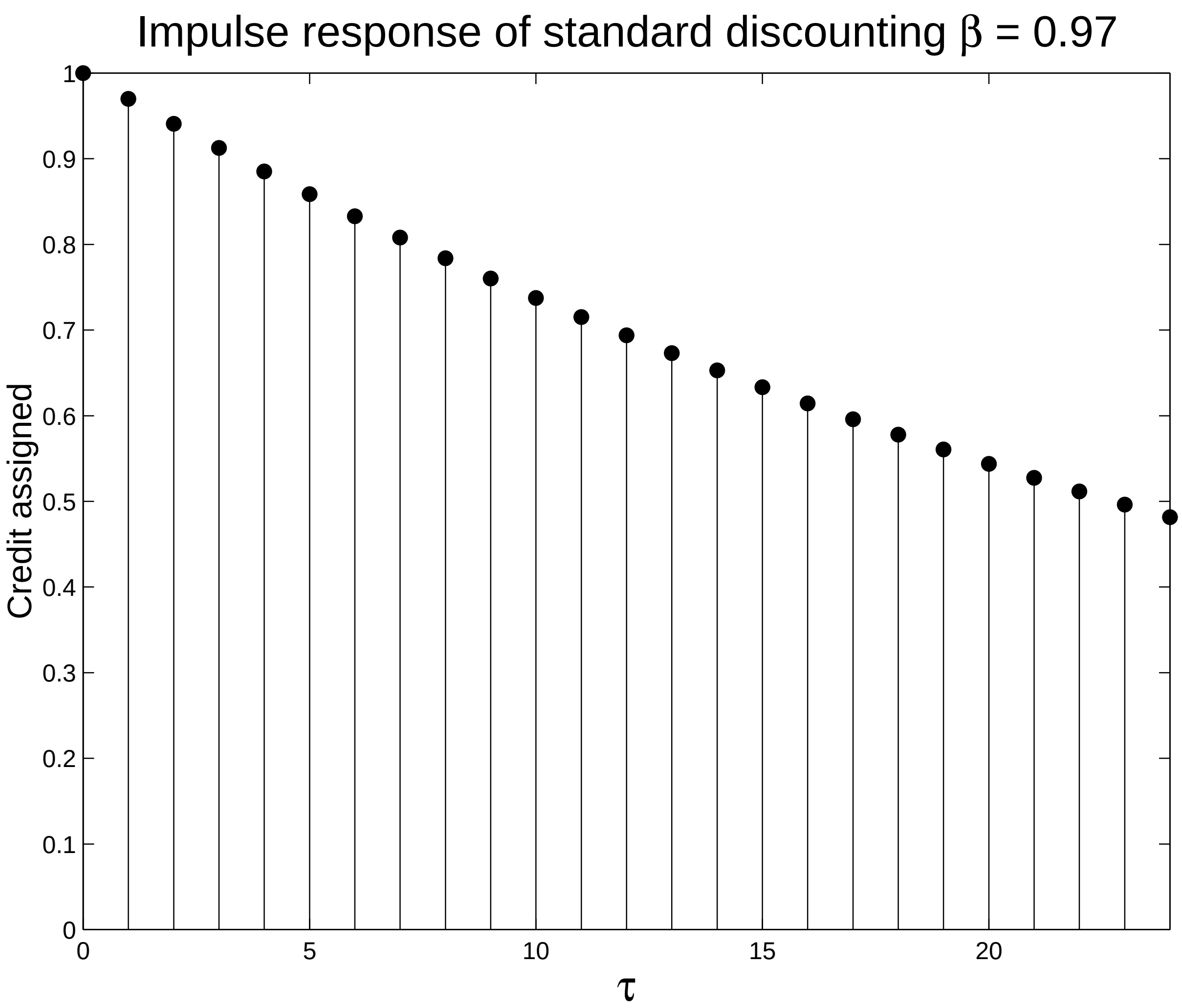
$$\widehat{\nabla_T \eta} = \frac{1}{T} \sum_{t=0}^{T-1} \frac{\nabla \mu(u_t|\theta, y_t)}{\mu(u_t|\theta, y_t)} \sum_{s=t+1}^{T} \beta^{s-t-1} r_s.$$

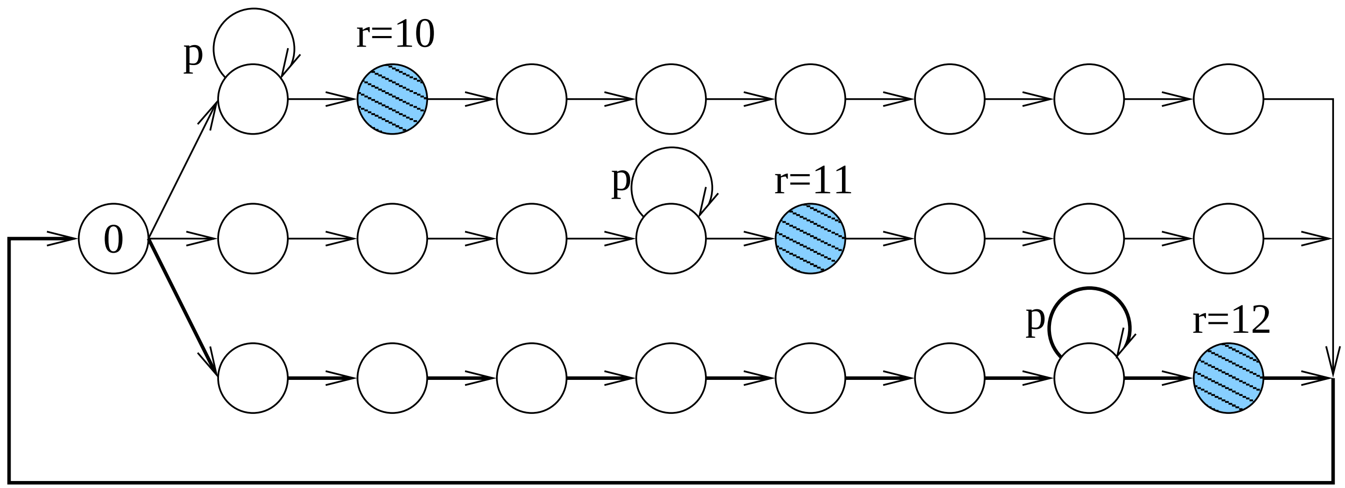
$$\Downarrow$$

$$z_{t+1} = \beta z_t + \frac{\nabla \mu(u_t|\theta, y_t)}{\mu(u_t|\theta, y_t)}$$

$$\widehat{\nabla_{t+1} \eta} = \widehat{\nabla_t \eta} + \frac{1}{t+1} [r_t z_{t+1} - \widehat{\nabla_t \eta}]$$

Standard discounting





We know the minimum delays from key action until rewards are issued.

Alternative filter I

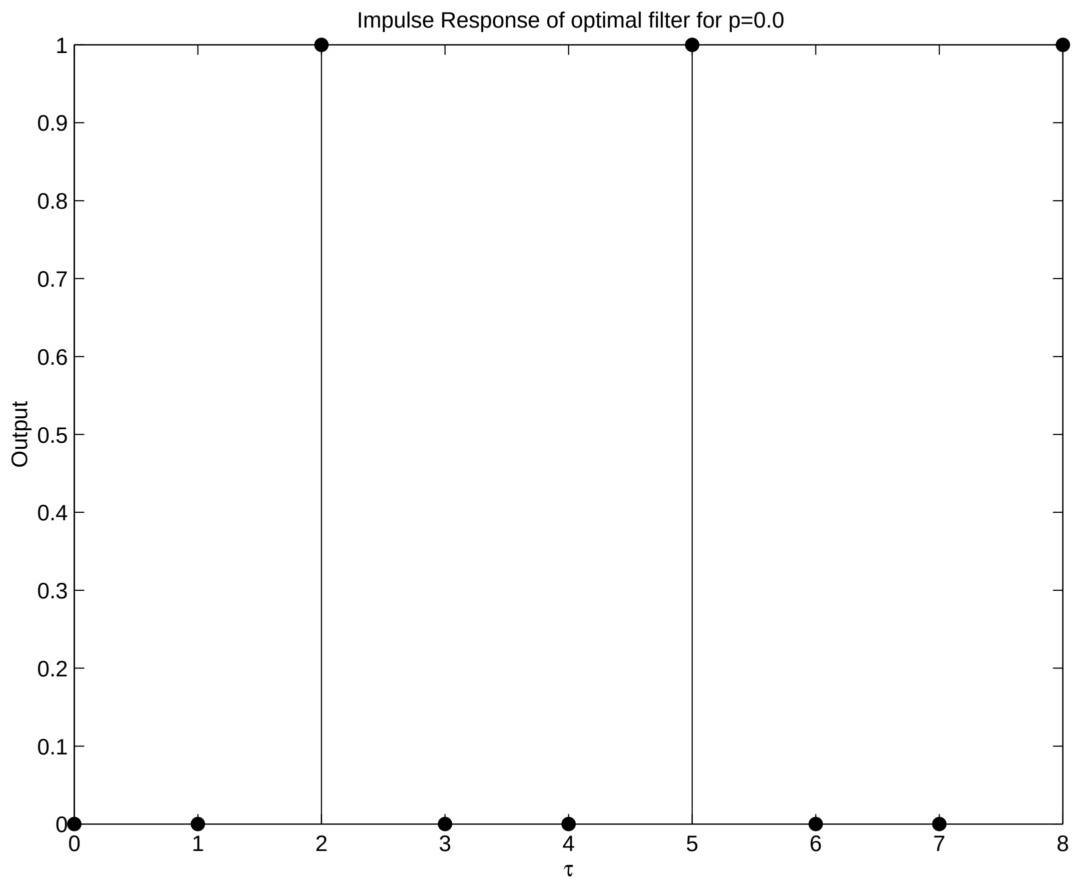


Figure 8: A bias optimal FIR filter for $P = 0$.

Alternative filter II

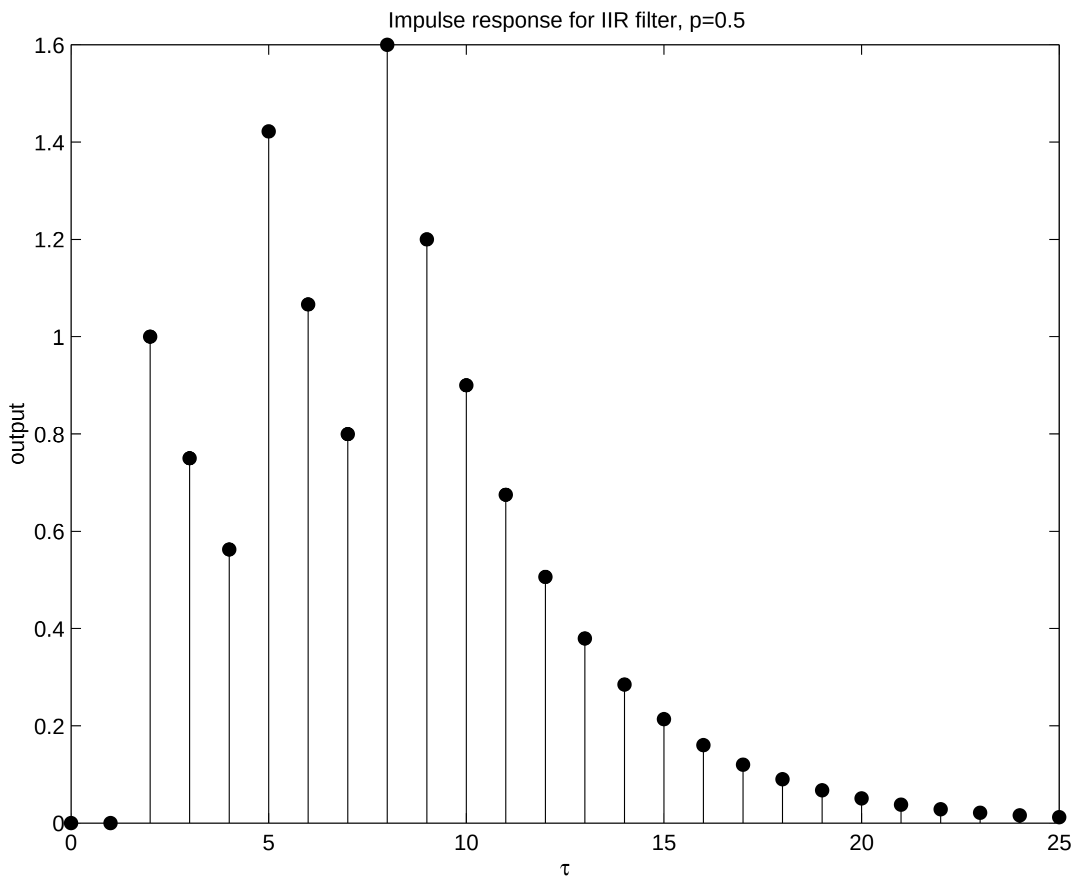


Figure 9: A "good" IIR filter for $P = 0.5$.

Arbitrary IIR Trace Filter

$$z_{t+1} = \beta z_t + \frac{\nabla \mu(u_t|\theta, y_t)}{\mu(u_t|\theta, y_t)}$$

$$\Downarrow$$

$$z_{t+1} = -\left(\sum_{i=1}^{|A|-1} A_i z_{t+1-i}\right) + \left(\sum_{i=0}^{|B|-1} B_i \frac{\nabla \mu(u_t|\theta, y_t)}{\mu(u_t|\theta, y_t)}\right).$$

Results

Trace type	Test I $p = 0$		Test II $p = 0.5$	
	Bias	var	Bias	var
$\beta = 0.9$	$176°$	12.3	$176°$	18.4
$\beta = 0.99$	$14.7°$	2090	$14.7°$	2140
FIR	$0.107°$	7.72	$4.35°$	59.5
IIR			$13.9°$	10.71

Table 1: Results of eligibility trace filtering tests. Note reduced variance of the filtered traces.

Key Conclusions

- I It is possible to perform a search for the optimal policy graph directly.
- II RL algorithms can be extended with I-states to perform this search.
- III A tough problem has been solved, using the sparse initialization trick to avoid the problem of low initial gradients.
- IV We can use eligibility trace filtering to add prior knowledge and hence reduce the gradient estimate variance.

Future Work

- I-state `GPOMDP` for larger problems from the literature.
- I-state `GPOMDP` for speech processing.
- I-state trained using EM like algorithm.
- Bounds on policy error introduced by too few I-states.
- Automatic selection of n_b.

Acknowledgments

- Peter Bartlett
- Sebastian Thrun

Questions?

http://csl.anu.edu.au/~daa/research.html