Connections Between Inference and Control

CS 294-112: Deep Reinforcement Learning

Sergey Levine

Class Notes

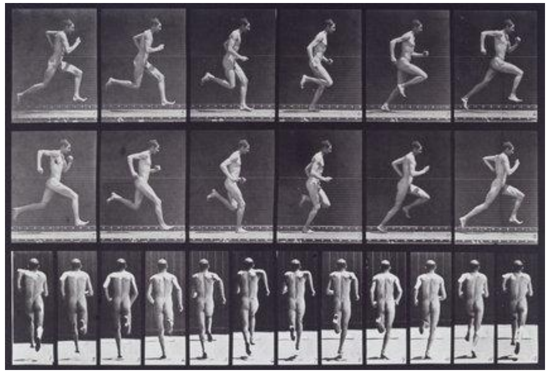
1. Homework 3 is due today, at 11:59 pm
2. Homework 4 comes out tonight
3. Final project proposal due on Monday!

Today’s Lecture

1. Does reinforcement learning and optimal control provide a reasonable model of human behavior?
2. Is there a better explanation?
3. Can we derive optimal control, reinforcement learning, and planning as *probabilistic inference*?
4. How does this change our RL algorithms?
5. (next week) We’ll see this is crucial for *inverse* reinforcement learning

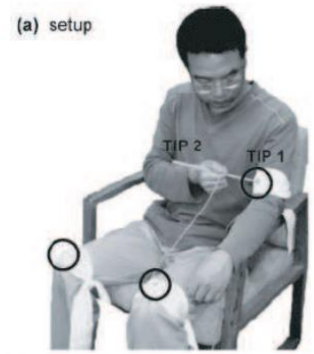
- Goals:
    - Understand the connection between inference and control
    - Understand how specific RL algorithms can be instantiated in this framework
    - Understand why this might be a good idea

Optimal Control as a Model of Human Behavior



Muybridge (c. 1870)

Mombaur et al. ‘09



Li & Todorov ‘06

Ziebart ‘08

$$\mathbf{a}_1, \ldots, \mathbf{a}_T = \arg \max_{\mathbf{a}_1, \ldots, \mathbf{a}_T} \sum_{t=1}^{T} r(\mathbf{s}_t, \mathbf{a}_t)$$

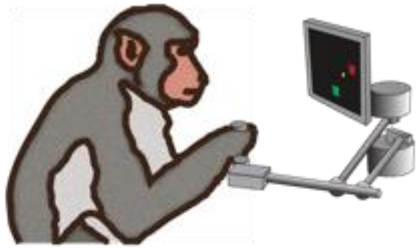
$$\mathbf{s}_{t+1} = f(\mathbf{s}_t, \mathbf{a}_t)$$

optimize this to explain the data

$$\pi = \arg \max_{\pi} E_{\mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t), \mathbf{a}_t \sim \pi(\mathbf{a}_t|\mathbf{s}_t)}[r(\mathbf{s}_t, \mathbf{a}_t)]$$

$$\mathbf{a}_t \sim \pi(\mathbf{a}_t|\mathbf{s}_t)$$

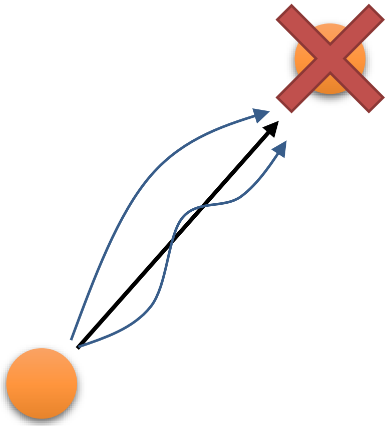
What if the data is **not** optimal?



some mistakes matter more than others!

behavior is **stochastic**

but good behavior is still the most likely



A probabilistic graphical model of decision making

$$\mathbf{a}_1, \ldots, \mathbf{a}_T = \arg \max_{\mathbf{a}_1, \ldots, \mathbf{a}_T} \sum_{t=1}^{T} r(\mathbf{s}_t, \mathbf{a}_t)$$

$$\mathbf{s}_{t+1} = f(\mathbf{s}_t, \mathbf{a}_t)$$

$$\underbrace{p(\mathbf{s}_{1:T}, \mathbf{a}_{1:T})}_{\tau} = ??$$

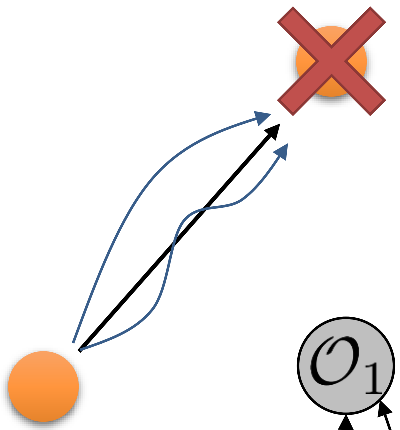
no assumption of optimal behavior!

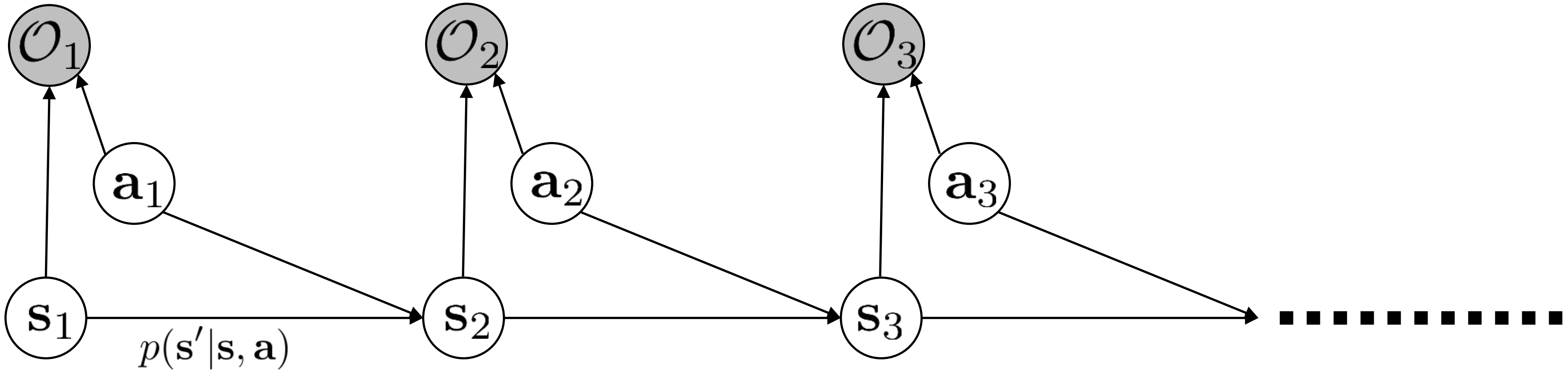
$$p(\tau | \mathcal{O}_{1:T})$$

$$p(\mathcal{O}_t | \mathbf{s}_t, \mathbf{a}_t) \propto \exp(r(\mathbf{s}_t, \mathbf{a}_t))$$

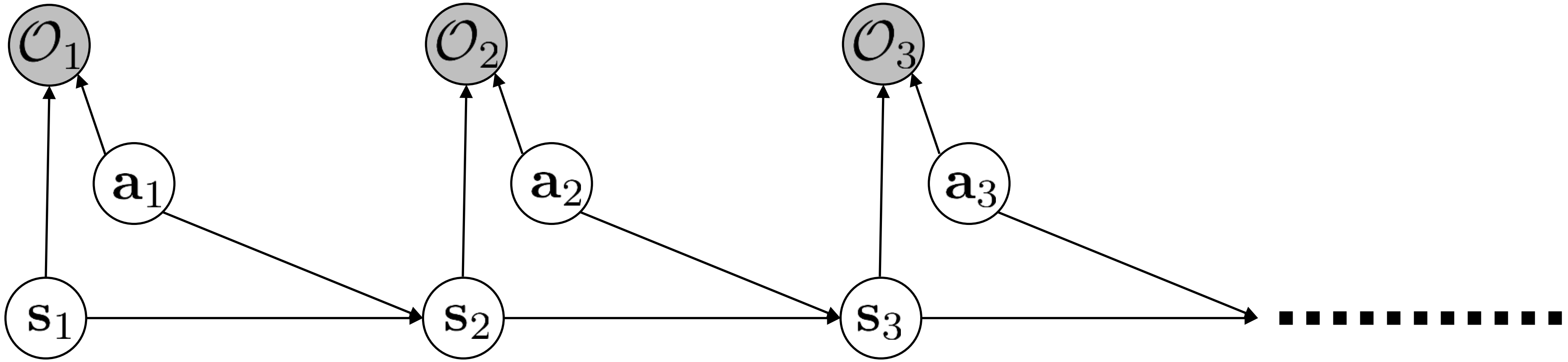
$$p(\tau | \mathcal{O}_{1:T}) = \frac{p(\tau, \mathcal{O}_{1:T})}{p(\mathcal{O}_{1:T})}$$

$$\propto p(\tau) \prod_t \exp(r(\mathbf{s}_t, \mathbf{a}_t)) = p(\tau) \exp\left(\sum_t r(\mathbf{s}_t, \mathbf{a}_t)\right)$$



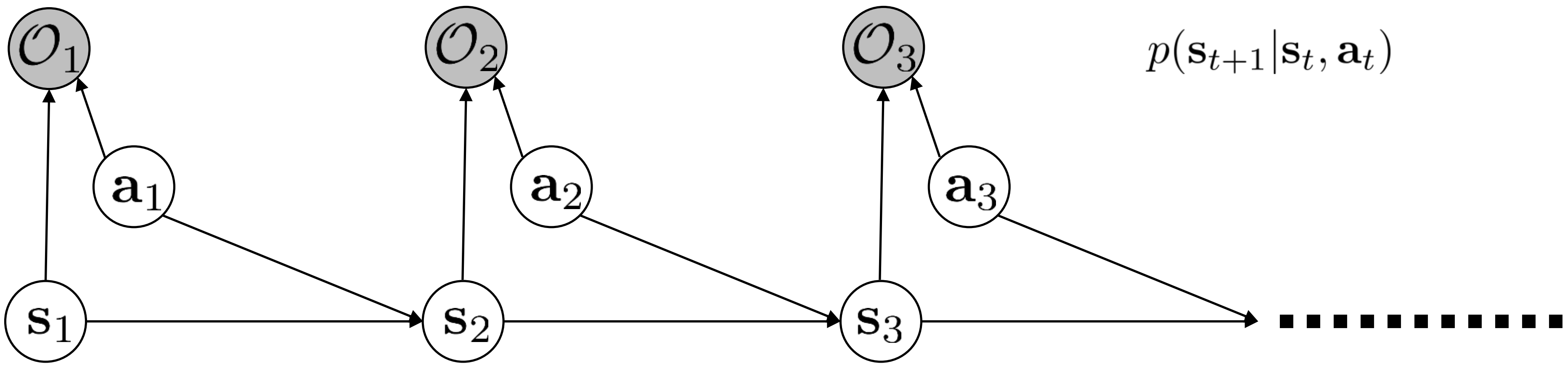


Why is this interesting?



- Can model suboptimal behavior (important for inverse RL)
- Can apply inference algorithms to solve control and planning problems
- Provides an explanation for why stochastic behavior might be preferred (useful for exploration and transfer learning)

Inference = planning



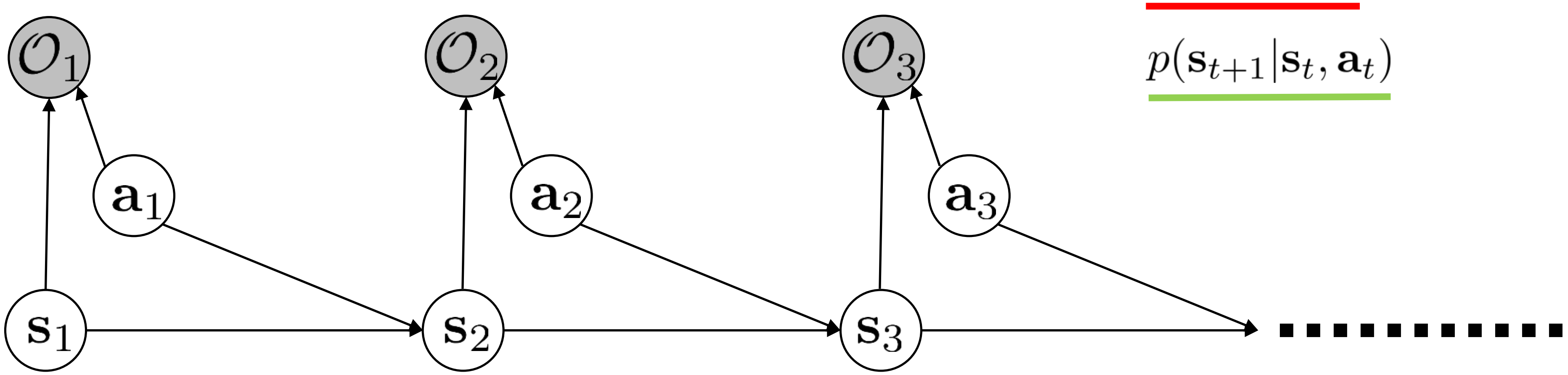
$p(\mathcal{O}_t|\mathbf{s}_t, \mathbf{a}_t) \propto \exp(r(\mathbf{s}_t, \mathbf{a}_t))$

$p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$

how to do inference?

1. compute backward messages $\beta_t(\mathbf{s}_t, \mathbf{a}_t) = p(\mathcal{O}_{t:T}|\mathbf{s}_t, \mathbf{a}_t)$
2. compute policy $p(\mathbf{a}_t|\mathbf{s}_t, \mathcal{O}_{1:T})$
3. compute forward messages $\alpha_t(\mathbf{s}_t) = p(\mathbf{s}_t|\mathcal{O}_{1:t-1})$

Backward messages



$p(\mathcal{O}_t|\mathbf{s}_t, \mathbf{a}_t) \propto \exp(r(\mathbf{s}_t, \mathbf{a}_t))$

$p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$

$$\beta_t(\mathbf{s}_t, \mathbf{a}_t) = p(\mathcal{O}_{t:T}|\mathbf{s}_t, \mathbf{a}_t)$$
$$= \int p(\mathcal{O}_{t:T}, \mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) d\mathbf{s}_{t+1}$$
$$= \int p(\mathcal{O}_{t+1:T}|\mathbf{s}_{t+1}) p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) p(\mathcal{O}_t|\mathbf{s}_t, \mathbf{a}_t) d\mathbf{s}_{t+1}$$

$$p(\mathcal{O}_{t+1:T}|\mathbf{s}_{t+1}) = \int p(\mathcal{O}_{t+1:T}|\mathbf{s}_{t+1}, \mathbf{a}_{t+1}) p(\mathbf{a}_{t+1}|\mathbf{s}_{t+1}) d\mathbf{a}_{t+1}$$

$\beta_t(\mathbf{s}_{t+1}, \mathbf{a}_{t+1})$

which actions are likely *a priori*
(assume uniform for now)

for $t = T-1$ to 1:

$$\beta_t(\mathbf{s}_t, \mathbf{a}_t) = p(\mathcal{O}_t|\mathbf{s}_t, \mathbf{a}_t) E_{\mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)}[\beta_{t+1}(\mathbf{s}_{t+1})]$$
$$\beta_t(\mathbf{s}_t) = E_{\mathbf{a}_t \sim p(\mathbf{a}_t|\mathbf{s}_t)}[\beta_t(\mathbf{s}_t, \mathbf{a}_t)]$$

A closer look at the backward pass

for $t = T-1$ to 1:

$$\beta_t(\mathbf{s}_t, \mathbf{a}_t) = p(\mathcal{O}_t|\mathbf{s}_t, \mathbf{a}_t) E_{\mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)}[\beta_{t+1}(\mathbf{s}_{t+1})]$$

$$\beta_t(\mathbf{s}_t) = E_{\mathbf{a}_t \sim p(\mathbf{a}_t|\mathbf{s}_t)}[\beta_t(\mathbf{s}_t, \mathbf{a}_t)]$$

value iteration algorithm:

1. set $Q(\mathbf{s}, \mathbf{a}) \leftarrow r(\mathbf{s}, \mathbf{a}) + \gamma E[V(\mathbf{s}')]$
2. set $V(\mathbf{s}) \leftarrow \max_{\mathbf{a}} Q(\mathbf{s}, \mathbf{a})$

let $V_t(\mathbf{s}_t) = \log \beta_t(\mathbf{s}_t)$

let $Q_t(\mathbf{s}_t, \mathbf{a}_t) = \log \beta_t(\mathbf{s}_t, \mathbf{a}_t)$

$$V_t(\mathbf{s}_t) = \log \int \exp(Q_t(\mathbf{s}_t, \mathbf{a}_t)) d\mathbf{a}_t$$

$V_t(\mathbf{s}_t) \rightarrow \max_{\mathbf{a}_t} Q_t(\mathbf{s}_t, \mathbf{a}_t)$ as $Q_t(\mathbf{s}_t, \mathbf{a}_t)$ gets bigger!

"optimistic" transition (not a good idea!)

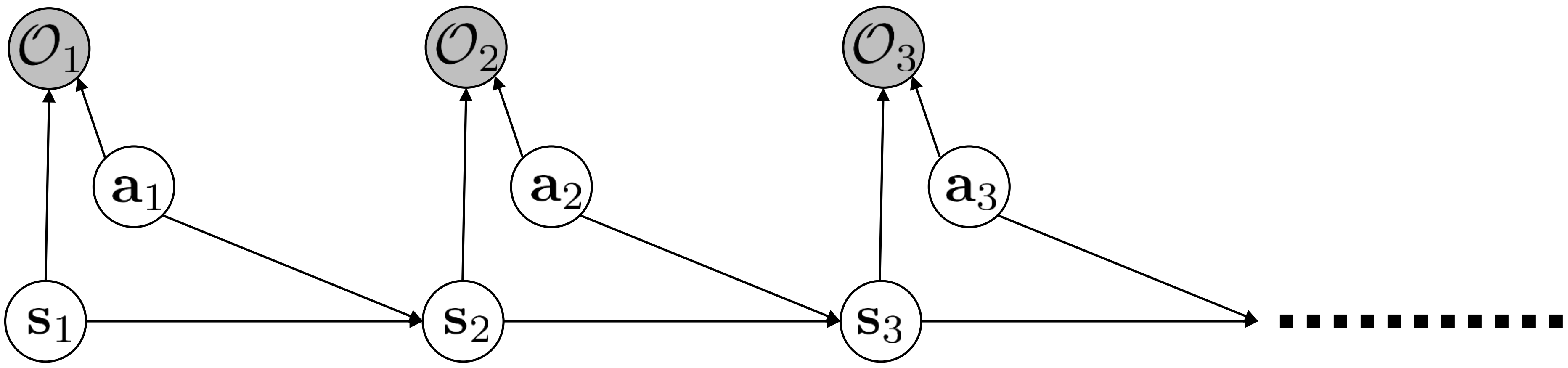
$$Q_t(\mathbf{s}_t, \mathbf{a}_t) = r(\mathbf{s}_t, \mathbf{a}_t) + \log E[\exp(V_{t+1}(\mathbf{s}_{t+1}))]$$

deterministic transition: $Q_t(\mathbf{s}_t, \mathbf{a}_t) = r(\mathbf{s}_t, \mathbf{a}_t) + V_{t+1}(\mathbf{s}_{t+1})$

a better stochastic model: $Q_t(\mathbf{s}_t, \mathbf{a}_t) = r(\mathbf{s}_t, \mathbf{a}_t) + E[V_{t+1}(\mathbf{s}_{t+1})]$

Ziebart et al. '10 "Modeling Interaction via the Principle of Maximum Causal Entropy"

Backward pass summary



$\beta_t(\mathbf{s}_t, \mathbf{a}_t) = p(\mathcal{O}_{t:T}|\mathbf{s}_t, \mathbf{a}_t)$

probability that we can be optimal at steps t through T given that we take action $\mathbf{a}_t$ in state $\mathbf{s}_t$

for $t = T - 1$ to 1:

$\beta_t(\mathbf{s}_t, \mathbf{a}_t) = p(\mathcal{O}_t|\mathbf{s}_t, \mathbf{a}_t) E_{\mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)}[\beta_{t+1}(\mathbf{s}_{t+1})]$

compute recursively from $t = T$ to $t = 1$

$\beta_t(\mathbf{s}_t) = E_{\mathbf{a}_t \sim p(\mathbf{a}_t|\mathbf{s}_t)}[\beta_t(\mathbf{s}_t, \mathbf{a}_t)]$

let $V_t(\mathbf{s}_t) = \log \beta_t(\mathbf{s}_t)$

let $Q_t(\mathbf{s}_t, \mathbf{a}_t) = \log \beta_t(\mathbf{s}_t, \mathbf{a}_t)$

log of β_t is "Q-function-like"

The action prior

remember this?

$$p(\mathcal{O}_{t+1:T}|\mathbf{s}_{t+1}) = \int p(\mathcal{O}_{t+1:T}|\mathbf{s}_{t+1}, \mathbf{a}_{t+1}) p(\mathbf{a}_{t+1}|\mathbf{s}_{t+1}) d\mathbf{a}_{t+1}$$

$\beta_t(\mathbf{s}_{t+1}, \mathbf{a}_{t+1})$

("soft max")

what if the action prior is not uniform?

$$V(\mathbf{s}_t) = \log \int \exp(Q(\mathbf{s}_t, \mathbf{a}_t) + \log p(\mathbf{a}_t|\mathbf{s}_t))\mathbf{a}_t$$

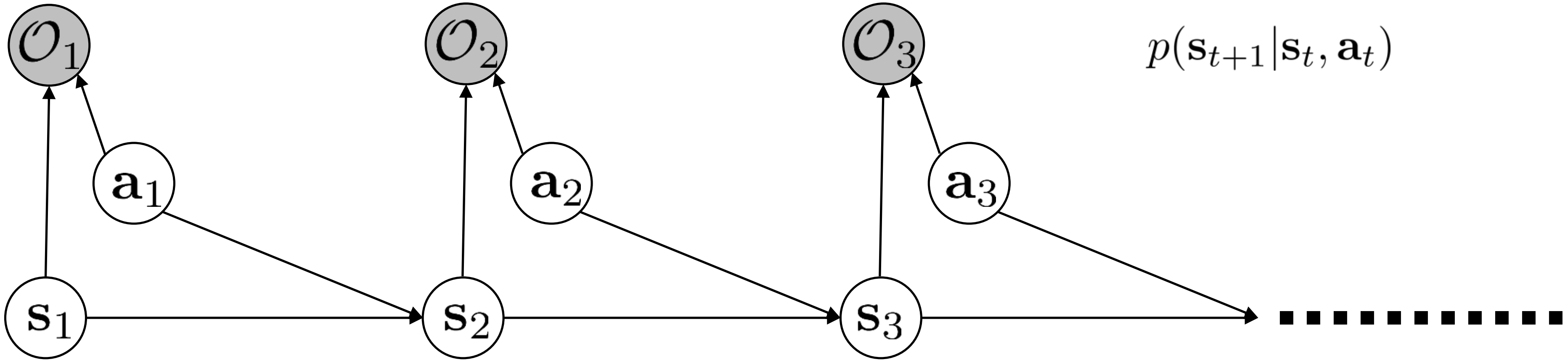
$$Q(\mathbf{s}_t, \mathbf{a}_t) = r(\mathbf{s}_t, \mathbf{a}_t) + E[V(\mathbf{s}_{t+1})]$$

let $\tilde{Q}(\mathbf{s}_t, \mathbf{a}_t) = r(\mathbf{s}_t, \mathbf{a}_t) + \log p(\mathbf{a}_t|\mathbf{s}_t) + E[V(\mathbf{s}_{t+1})]$

$$V(\mathbf{s}_t) = \log \int \exp(\tilde{Q}(\mathbf{s}_t, \mathbf{a}_t))\mathbf{a}_t \quad \Leftrightarrow \quad V(\mathbf{s}_t) = \log \int \exp(Q(\mathbf{s}_t, \mathbf{a}_t) + \log p(\mathbf{a}_t|\mathbf{s}_t))\mathbf{a}_t$$

can **always** fold the action prior into the reward! uniform action prior can be assumed without loss of generality

Policy computation



$$p(\mathcal{O}_t|\mathbf{s}_t,\mathbf{a}_t) \propto \exp(r(\mathbf{s}_t,\mathbf{a}_t))$$

$$p(\mathbf{s}_{t+1}|\mathbf{s}_t,\mathbf{a}_t)$$

2. compute policy $p(\mathbf{a}_t|\mathbf{s}_t,\mathcal{O}_{1:T})$

$$\beta_t(\mathbf{s}_t,\mathbf{a}_t) = p(\mathcal{O}_{t:T}|\mathbf{s}_t,\mathbf{a}_t)$$

$$\beta_t(\mathbf{s}_t) = p(\mathcal{O}_{t:T}|\mathbf{s}_t)$$

$$\begin{aligned}
p(\mathbf{a}_t|\mathbf{s}_t,\mathcal{O}_{1:T}) &= \pi(\mathbf{a}_t|\mathbf{s}_t) \\
&= p(\mathbf{a}_t|\mathbf{s}_t,\mathcal{O}_{t:T}) \\
&= \frac{p(\mathbf{a}_t,\mathbf{s}_t|\mathcal{O}_{t:T})}{p(\mathbf{s}_t|\mathcal{O}_{t:T})} \\
&= \frac{p(\mathcal{O}_{t:T}|\mathbf{a}_t,\mathbf{s}_t)p(\mathbf{a}_t,\mathbf{s}_t)/\cancel{p(\mathcal{O}_{t:T})}}{p(\mathcal{O}_{t:T}|\mathbf{s}_t)p(\mathbf{s}_t)/\cancel{p(\mathcal{O}_{t:T})}} \\
&= \frac{p(\mathcal{O}_{t:T}|\mathbf{a}_t,\mathbf{s}_t)}{p(\mathcal{O}_{t:T}|\mathbf{s}_t)}\frac{p(\mathbf{a}_t,\mathbf{s}_t)}{p(\mathbf{s}_t)} = \frac{\beta_t(\mathbf{s}_t,\mathbf{a}_t)}{\beta_t(\mathbf{s}_t)}\cancel{p(\mathbf{a}_t|\mathbf{s}_t)}
\end{aligned}$$

$$\pi(\mathbf{a}_t|\mathbf{s}_t) = \frac{\beta_t(\mathbf{s}_t,\mathbf{a}_t)}{\beta_t(\mathbf{s}_t)}$$

Policy computation with value functions

for $t = T-1$ to 1:

$$Q_t(\mathbf{s}_t, \mathbf{a}_t) = r(\mathbf{s}_t, \mathbf{a}_t) + E[V_{t+1}(\mathbf{s}_{t+1})]$$

$$V_t(\mathbf{s}_t) = \log \int \exp(Q_t(\mathbf{s}_t, \mathbf{a}_t))\mathbf{a}_t$$

$$\pi(\mathbf{a}_t|\mathbf{s}_t) = \frac{\beta_t(\mathbf{s}_t, \mathbf{a}_t)}{\beta_t(\mathbf{s}_t)}$$

$$V_t(\mathbf{s}_t) = \log \beta_t(\mathbf{s}_t)$$

$$Q_t(\mathbf{s}_t, \mathbf{a}_t) = \log \beta_t(\mathbf{s}_t, \mathbf{a}_t)$$

$$\pi(\mathbf{a}_t|\mathbf{s}_t) = \exp(Q_t(\mathbf{s}_t, \mathbf{a}_t) - V_t(\mathbf{s}_t)) = \exp(A_t(\mathbf{s}_t, \mathbf{a}_t))$$

variants:

discounted SOC: $Q_t(\mathbf{s}_t, \mathbf{a}_t) = r(\mathbf{s}_t, \mathbf{a}_t) + \gamma E[V_{t+1}(\mathbf{s}_{t+1})]$

explicit temperature: $V_t(\mathbf{s}_t) = \alpha \log \int \exp\left(\frac{1}{\alpha} Q_t(\mathbf{s}_t, \mathbf{a}_t)\right) d\mathbf{a}_t$

Policy computation summary

$$\pi(\mathbf{a}_t|\mathbf{s}_t) = \exp(Q_t(\mathbf{s}_t, \mathbf{a}_t) - V_t(\mathbf{s}_t)) = \exp(A_t(\mathbf{s}_t, \mathbf{a}_t))$$

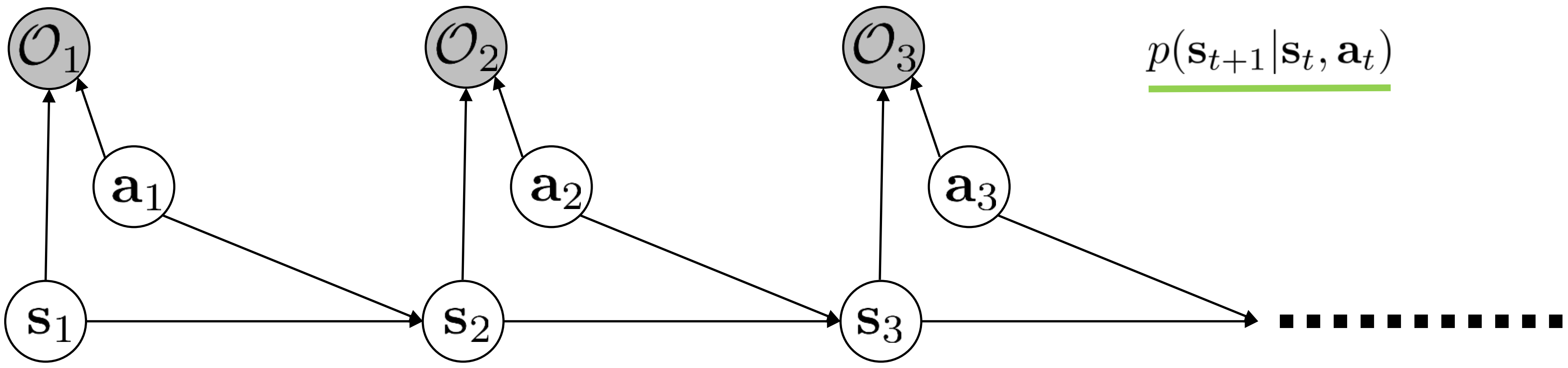
$$\text{with temperature: } \pi(\mathbf{a}_t|\mathbf{s}_t) = \exp(\tfrac{1}{\alpha}Q_t(\mathbf{s}_t, \mathbf{a}_t) - \tfrac{1}{\alpha}V_t(\mathbf{s}_t)) = \exp(\tfrac{1}{\alpha}A_t(\mathbf{s}_t, \mathbf{a}_t))$$

- Natural interpretation: better actions are more probable
- Random tie-breaking
- Analogous to Boltzmann exploration
- Approaches greedy policy as temperature decreases

Forward messages

$$p(\mathcal{O}_t|\mathbf{s}_t, \mathbf{a}_t) \propto \exp(r(\mathbf{s}_t, \mathbf{a}_t))$$

$$p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$$



$$\alpha_t(\mathbf{s}_t) = p(\mathbf{s}_t|\mathcal{O}_{1:t-1})$$

$$= \int \underbrace{p(\mathbf{s}_t|\mathbf{s}_{t-1}, \mathbf{a}_{t-1})}p(\mathbf{a}_{t-1}|\mathbf{s}_{t-1}, \mathcal{O}_{t-1})\underbrace{p(\mathbf{s}_{t-1}|\mathcal{O}_{1:t-2})}_{\alpha_{t-1}(\mathbf{s}_{t-1})}d\mathbf{s}_{t-1}d\mathbf{a}_{t-1}$$

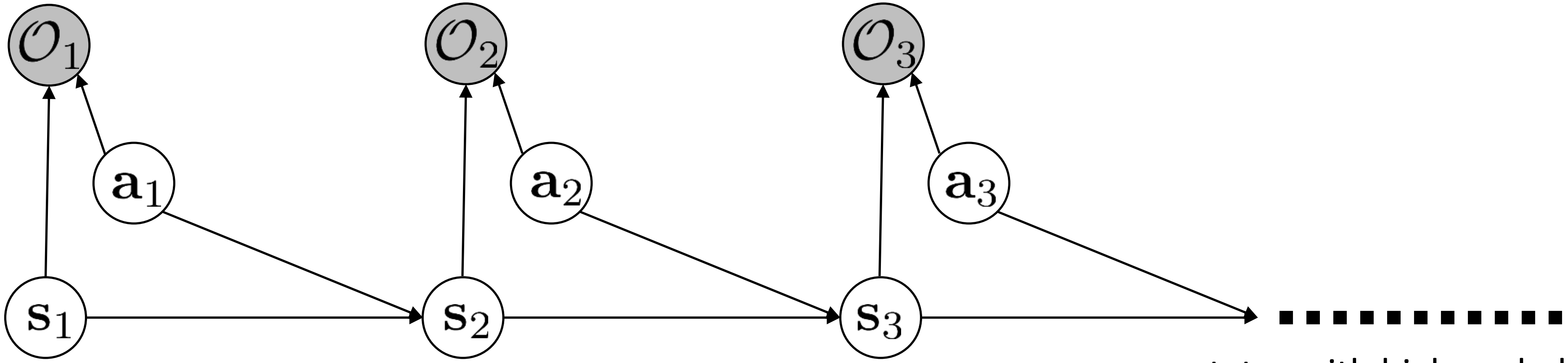
$\alpha_1(\mathbf{s}_1) = p(\mathbf{s}_1)$ (usually known)

$$p(\mathbf{a}_{t-1}|\mathbf{s}_{t-1}, \mathcal{O}_{t-1}) = \frac{p(\mathcal{O}_{t-1}|\mathbf{s}_{t-1}, \mathbf{a}_{t-1})p(\mathbf{a}_{t-1}|\mathbf{s}_{t-1})}{p(\mathcal{O}_{t-1}|\mathbf{s}_{t-1})}$$

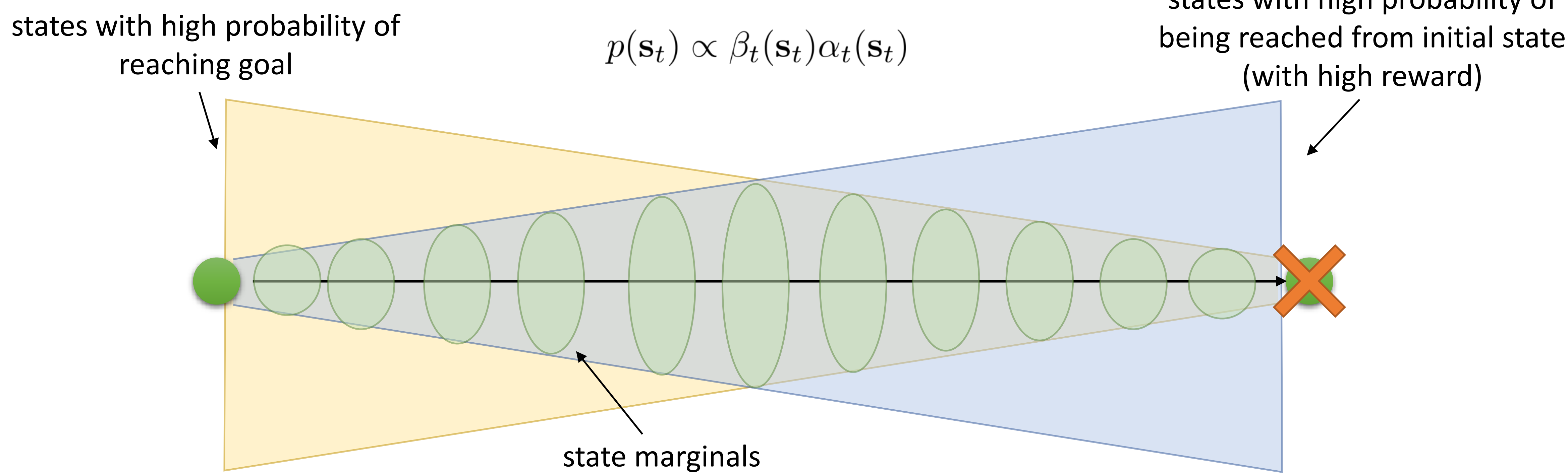
what if we want $p(\mathbf{s}_t|\mathcal{O}_{1:T})$?

$$p(\mathbf{s}_t|\mathcal{O}_{1:T}) = \frac{p(\mathbf{s}_t, \mathcal{O}_{1:T})}{p(\mathcal{O}_{1:T})} = \frac{\overbrace{p(\mathcal{O}_{t:T}|\mathbf{s}_t)}^{\beta_t(\mathbf{s}_t)}p(\mathbf{s}_t, \mathcal{O}_{1:t-1})}{p(\mathcal{O}_{1:T})} \propto \beta_t(\mathbf{s}_t)\underbrace{p(\mathbf{s}_t|\mathcal{O}_{1:t-1})}_{\alpha_t(\mathbf{s}_t)}\cancel{p(\mathcal{O}_{1:t-1})} \propto \beta_t(\mathbf{s}_t)\alpha_t(\mathbf{s}_t)$$

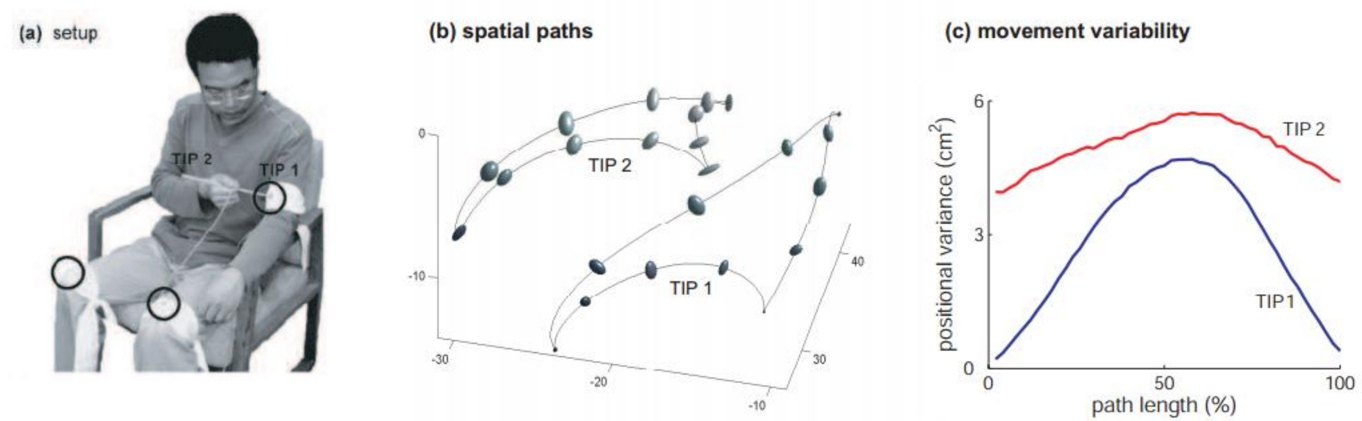
Forward/backward message intersection



$$p(\mathbf{s}_t) \propto \beta_t(\mathbf{s}_t)\alpha_t(\mathbf{s}_t)$$



Forward/backward message intersection

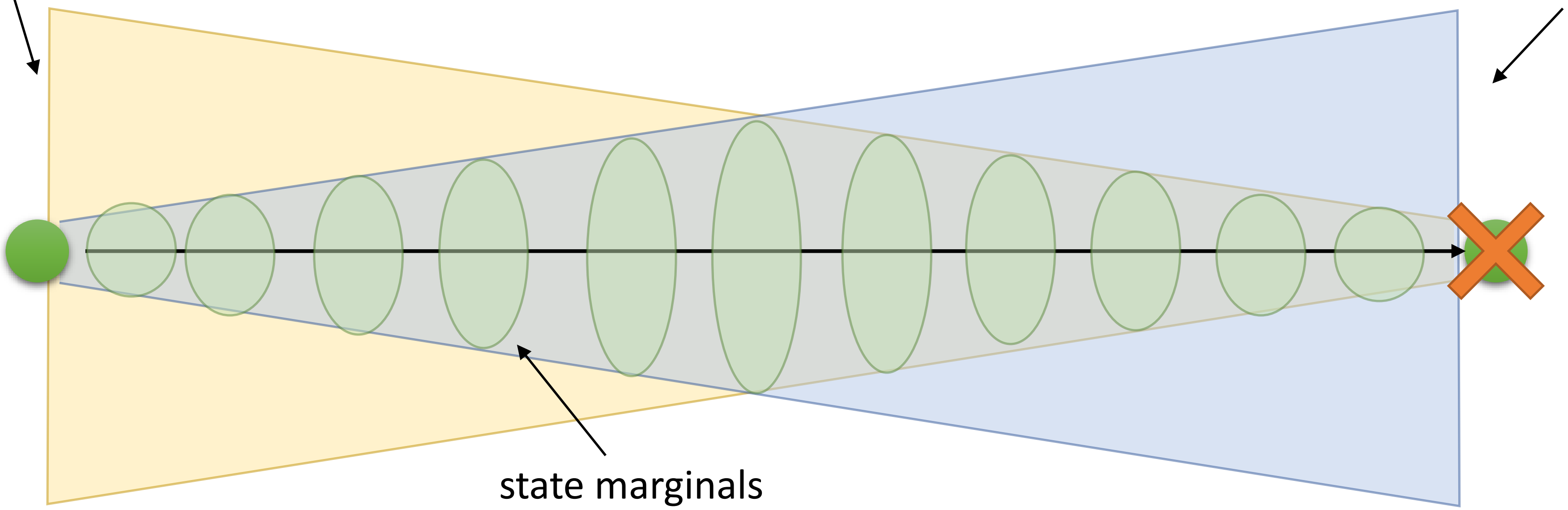


Li & Todorov, 2006

states with high probability of reaching goal

$$p(\mathbf{s}_t) \propto \beta_t(\mathbf{s}_t)\alpha_t(\mathbf{s}_t)$$

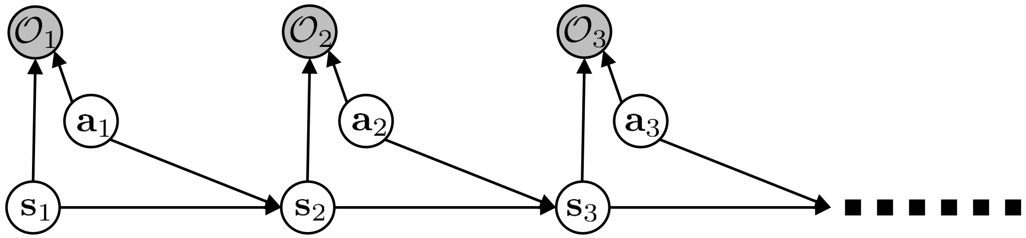
states with high probability of being reached from initial state (with high reward)



state marginals

Summary

1. Probabilistic graphical model for optimal control



2. Control = inference (similar to HMM, EKF, etc.)

3. Very similar to dynamic programming, value iteration, etc. (but “soft”)

Q-learning with soft optimality

standard Q-learning: $\phi \leftarrow \phi + \alpha \nabla_\phi Q_\phi(\mathbf{s}, \mathbf{a})(r(\mathbf{s}, \mathbf{a}) + \gamma V(\mathbf{s}') - Q_\phi(\mathbf{s}, \mathbf{a}))$

target value: $V(\mathbf{s}') = \max_{\mathbf{a}'} Q_\phi(\mathbf{s}', \mathbf{a}')$

soft Q-learning: $\phi \leftarrow \phi + \alpha \nabla_\phi Q_\phi(\mathbf{s}, \mathbf{a})(r(\mathbf{s}, \mathbf{a}) + \gamma V(\mathbf{s}') - Q_\phi(\mathbf{s}, \mathbf{a}))$

target value: $V(\mathbf{s}') = \text{soft}\max_{\mathbf{a}'} Q_\phi(\mathbf{s}', \mathbf{a}') = \log \int \exp(Q_\phi(\mathbf{s}', \mathbf{a}')) d\mathbf{a}'$

$\pi(\mathbf{a}|\mathbf{s}) = \exp(Q_\phi(\mathbf{s}, \mathbf{a}) - V(\mathbf{s})) = \exp(A(\mathbf{s}, \mathbf{a}))$

1. take some action $\mathbf{a}_i$ and observe $(\mathbf{s}_i, \mathbf{a}_i, \mathbf{s}'_i, r_i)$, add it to $\mathcal{R}$
2. sample mini-batch $\{\mathbf{s}_j, \mathbf{a}_j, \mathbf{s}'_j, r_j\}$ from $\mathcal{R}$ uniformly
3. compute $y_j = r_j + \gamma \text{soft}\max_{\mathbf{a}'_j} Q_{\phi'}(\mathbf{s}'_j, \mathbf{a}'_j)$ using *target* network $Q_{\phi'}$
4. $\phi \leftarrow \phi - \alpha \sum_j \frac{dQ_\phi}{d\phi}(\mathbf{s}_j, \mathbf{a}_j)(Q_\phi(\mathbf{s}_j, \mathbf{a}_j) - y_j)$
5. update ϕ': copy ϕ every N steps, or Polyak average $\phi' \leftarrow \tau\phi' + (1-\tau)\phi$

Policy gradient with soft optimality

$$\pi(\mathbf{a}|\mathbf{s}) = \exp(Q_\phi(\mathbf{s},\mathbf{a}) - V(\mathbf{s})) \text{ optimizes } \sum_t E_{\pi(\mathbf{s}_t,\mathbf{a}_t)}[r(\mathbf{s}_t,\mathbf{a}_t)] + E_{\pi(\mathbf{s}_t)}[\mathcal{H}(\pi(\mathbf{a}_t|\mathbf{s}_t))]$$

policy entropy

intuition: $\pi(\mathbf{a}|\mathbf{s}) \propto \exp(Q_\phi(\mathbf{s},\mathbf{a}))$ when π minimizes $D_{\mathrm{KL}}(\pi(\mathbf{a}|\mathbf{s}) \| \frac{1}{Z}\exp(Q(\mathbf{s},\mathbf{a})))$

$$D_{\mathrm{KL}}(\pi(\mathbf{a}|\mathbf{s}) \| \tfrac{1}{Z}\exp(Q(\mathbf{s},\mathbf{a}))) = E_{\pi(\mathbf{a}|\mathbf{s})}[Q(\mathbf{s},\mathbf{a})] - \mathcal{H}(\pi)$$

often referred to as “entropy regularized” policy gradient

combats premature entropy collapse

turns out to be closely related to soft Q-learning:
see Haarnoja et al. ‘17 and Schulman et al. ‘17

Ziebart et al. ‘10 “Modeling Interaction via the Principle of Maximum Causal Entropy”

Policy gradient vs Q-learning

policy gradient derivation:

$$J(\theta) = \sum_t E_{\pi(\mathbf{s}_t,\mathbf{a}_t)}[r(\mathbf{s}_t,\mathbf{a}_t)] + E_{\pi(\mathbf{s}_t)}[\underbrace{\mathcal{H}(\pi(\mathbf{a}|\mathbf{s}_t))}_{E_{\pi(\mathbf{a}_t|\mathbf{s}_t)}[-\log\pi(\mathbf{a}_t|\mathbf{s}_t)]}] = \sum_t E_{\pi(\mathbf{s}_t,\mathbf{a}_t)}[r(\mathbf{s}_t,\mathbf{a}_t) - \log\pi(\mathbf{a}_t|\mathbf{s}_t)]$$

$$\nabla_\theta\left[\sum_t E_{\pi(\mathbf{s}_t,\mathbf{a}_t)}[r(\mathbf{s}_t,\mathbf{a}_t) - \log\pi(\mathbf{a}_t|\mathbf{s}_t)]\right]$$

$$\approx \frac{1}{N}\sum_i\sum_t \nabla_\theta\log\pi(\mathbf{a}_t|\mathbf{s}_t)\left(r(\mathbf{s}_t,\mathbf{a}_t) + \underbrace{\left(\sum_{t'=t+1}^{T} r(\mathbf{s}_{t'},\mathbf{a}_{t'}) - \log\pi(\mathbf{a}_{t'}|\mathbf{s}_{t'})\right)}_{\approx Q(\mathbf{s}_{t+1},\mathbf{a}_{t+1})} - \log\pi(\mathbf{a}_t|\mathbf{s}_t) - 1\right)$$

can ignore (baseline)

recall: $\log\pi(\mathbf{a}_t|\mathbf{s}_t) = Q(\mathbf{s}_t,\mathbf{a}_t) - V(\mathbf{s}_t)$

$$\approx \frac{1}{N}\sum_i\sum_t (\nabla_\theta Q(\mathbf{a}_t|\mathbf{s}_t) - \nabla_\theta V(\mathbf{s}_t))\left(r(\mathbf{s}_t,\mathbf{a}_t) + Q(\mathbf{s}_{t+1},\mathbf{a}_{t+1}) - Q(\mathbf{s}_t,\mathbf{a}_t) + V(\mathbf{s}_t)\right)$$

Q-learning: $$-\frac{1}{N}\sum_i\sum_t \nabla_\theta Q(\mathbf{a}_t|\mathbf{s}_t)\left(r(\mathbf{s}_t,\mathbf{a}_t) + \operatorname{soft}\max_{\mathbf{a}_{t+1}} Q(\mathbf{s}_{t+1},\mathbf{a}_{t+1}) - Q(\mathbf{s}_t,\mathbf{a}_t)\right)$$

descent (vs ascent)

off-policy correction

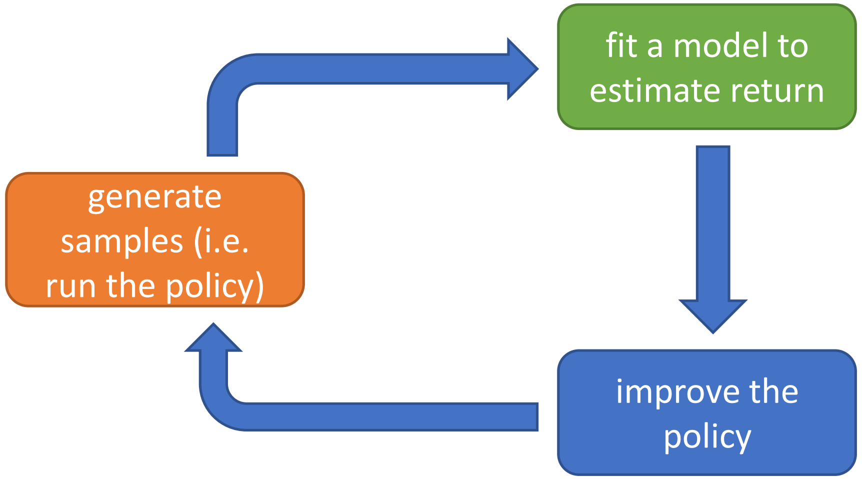
Benefits of soft optimality

- Improve exploration and prevent entropy collapse
- Easier to specialize (finetune) policies for more specific tasks
- Principled approach to break ties
- Better robustness (due to wider coverage of states)
- Can reduce to hard optimality as reward magnitude increases
- Good model for modeling human behavior (more on this later)

Review

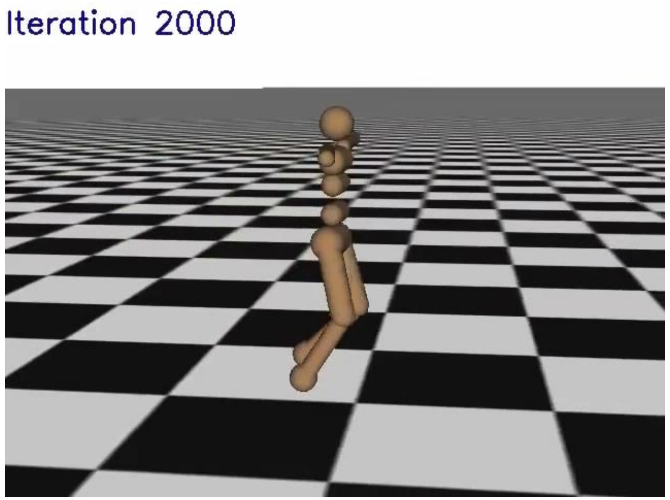
- Reinforcement learning can be viewed as inference in a graphical model
  - Value function is a backward message
  - Maximize reward and entropy (the bigger the rewards, the less entropy matters)
- Soft Q-learning
- Entropy-regularized policy gradient

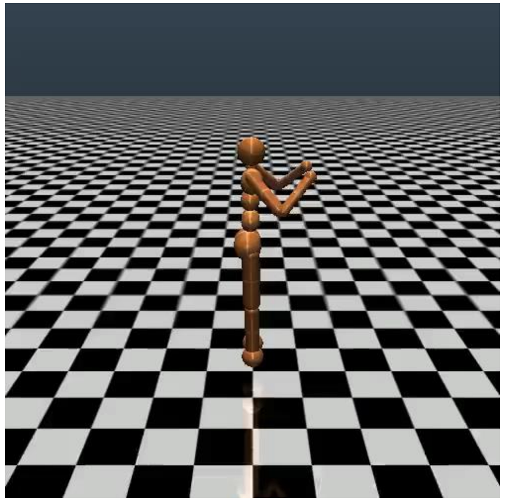
$$Q_\phi(\mathbf{s},\mathbf{a}) \leftarrow r(\mathbf{s},\mathbf{a}) + \gamma \text{soft max}_{\mathbf{a}'} Q_\phi(\mathbf{s}',\mathbf{a}')$$

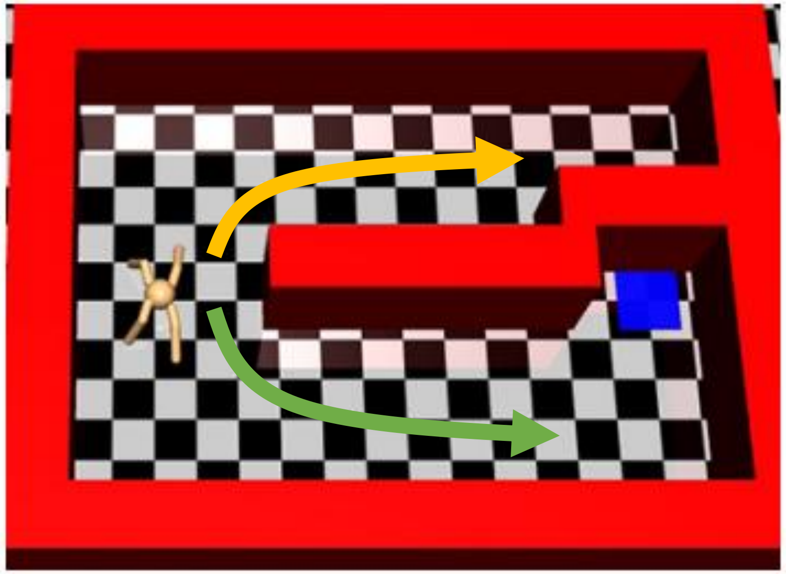


$$\mathbf{a} = \arg\max_{\mathbf{a}} Q_\phi(\mathbf{s},\mathbf{a})$$

Stochastic models for learning control



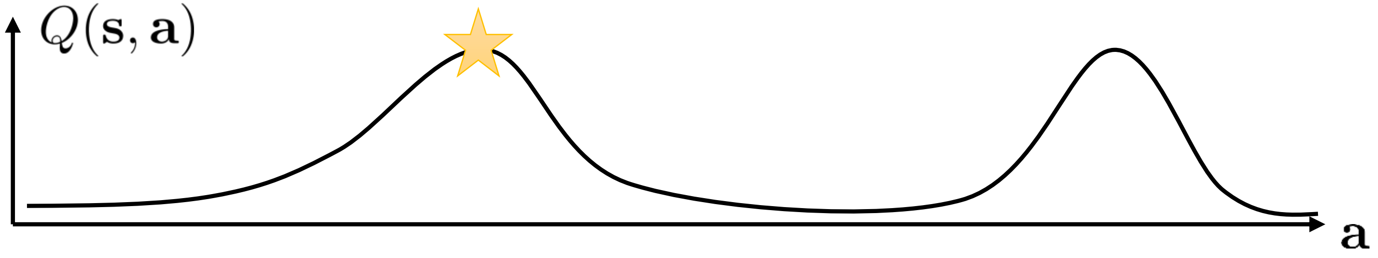


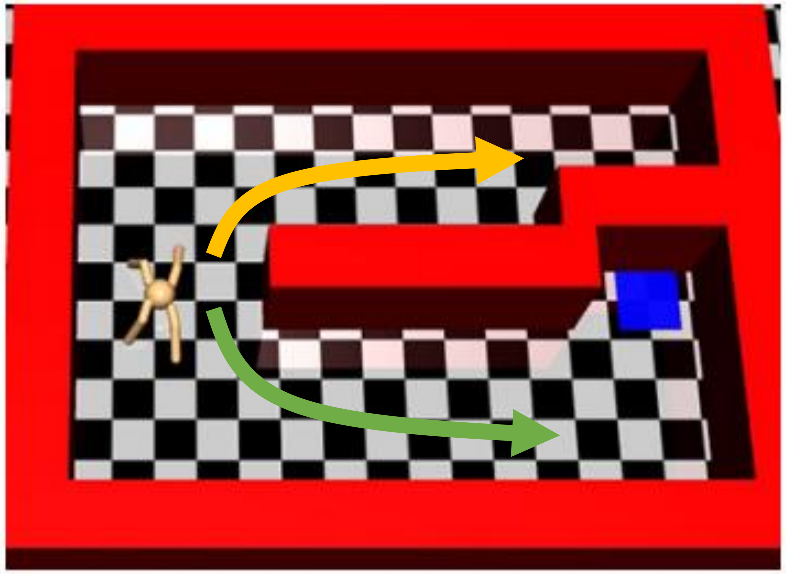


- How can we track *both* hypotheses?

Stochastic energy-based policies

Q-function: $Q(\mathbf{s},\mathbf{a}) : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$





$$\pi(\mathbf{a}|\mathbf{s}) \propto \exp(Q(\mathbf{s},\mathbf{a}))$$

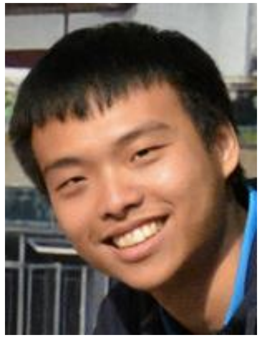
$$\pi(\mathbf{a}_t|\mathbf{s}_t) = \exp(Q_t(\mathbf{s}_t,\mathbf{a}_t) - V_t(\mathbf{s}_t)) = \exp(A_t(\mathbf{s}_t,\mathbf{a}_t))$$

$$Q_t(\mathbf{s}_t,\mathbf{a}_t) = r(\mathbf{s}_t,\mathbf{a}_t) + E[V_{t+1}(\mathbf{s}_{t+1})]$$

$$V_t(\mathbf{s}_t) = \log \int \exp(Q_t(\mathbf{s}_t,\mathbf{a}_t))\mathbf{a}_t$$

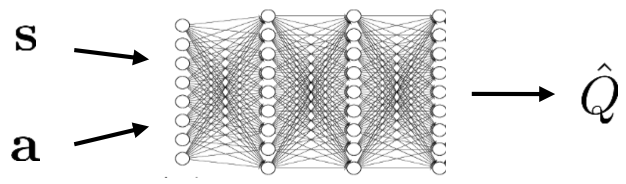
Tuomas Haarnoja

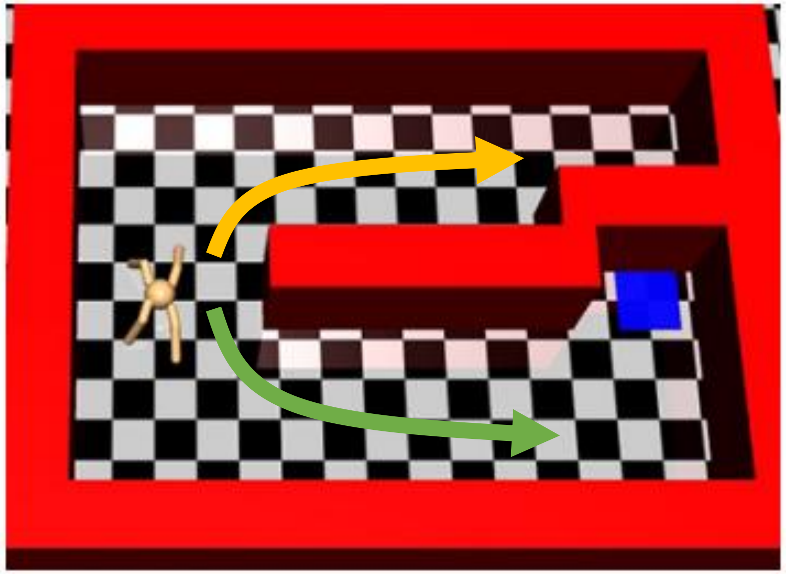
Haoran Tang



Soft Q-learning

Learned (neural network) Q-function: $Q_\theta(\mathbf{s}, \mathbf{a})$





Q-learning: $\theta \leftarrow \theta + \alpha \nabla_\theta Q_\theta(\mathbf{s}, \mathbf{a})(r(\mathbf{s}, \mathbf{a}) + \gamma V(\mathbf{s}') - Q_\theta(\mathbf{s}, \mathbf{a}))$

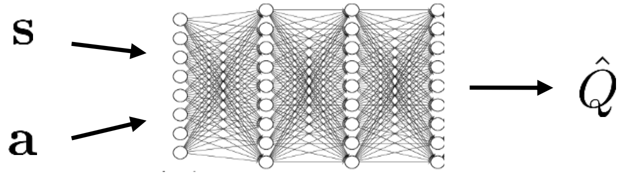
target value: $V(\mathbf{s}') = \max_{\mathbf{a}'} Q_\theta(\mathbf{s}', \mathbf{a}')$

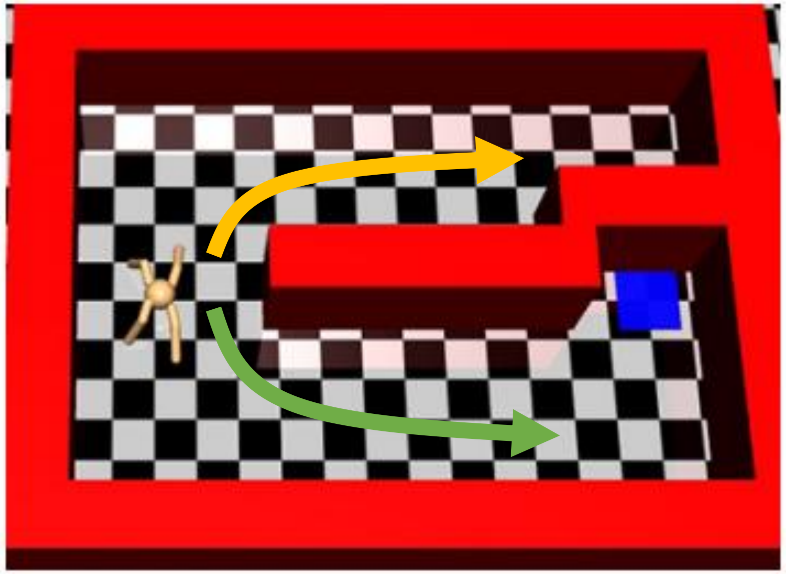
soft Q-learning: $\theta \leftarrow \theta + \alpha \nabla_\theta Q_\theta(\mathbf{s}, \mathbf{a})(r(\mathbf{s}, \mathbf{a}) + \gamma V(\mathbf{s}') - Q_\theta(\mathbf{s}, \mathbf{a}))$

target value: $V(\mathbf{s}') = \text{soft max}_{\mathbf{a}'} Q_\theta(\mathbf{s}', \mathbf{a}') = \log \int \exp(Q_\theta(\mathbf{s}', \mathbf{a}')) d\mathbf{a}'$

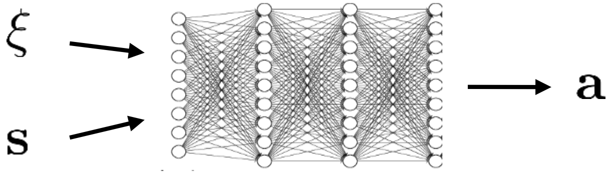
Tractable amortized inference for continuous actions

$\pi(\mathbf{a}|\mathbf{s}) \propto \exp(Q(\mathbf{s},\mathbf{a}))$





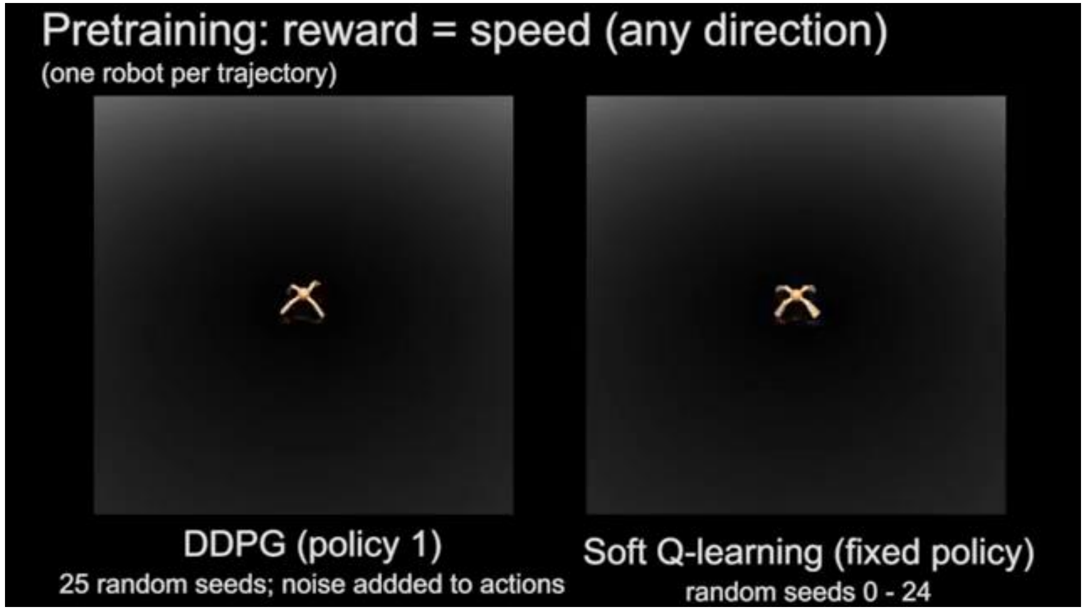
stochastic network:

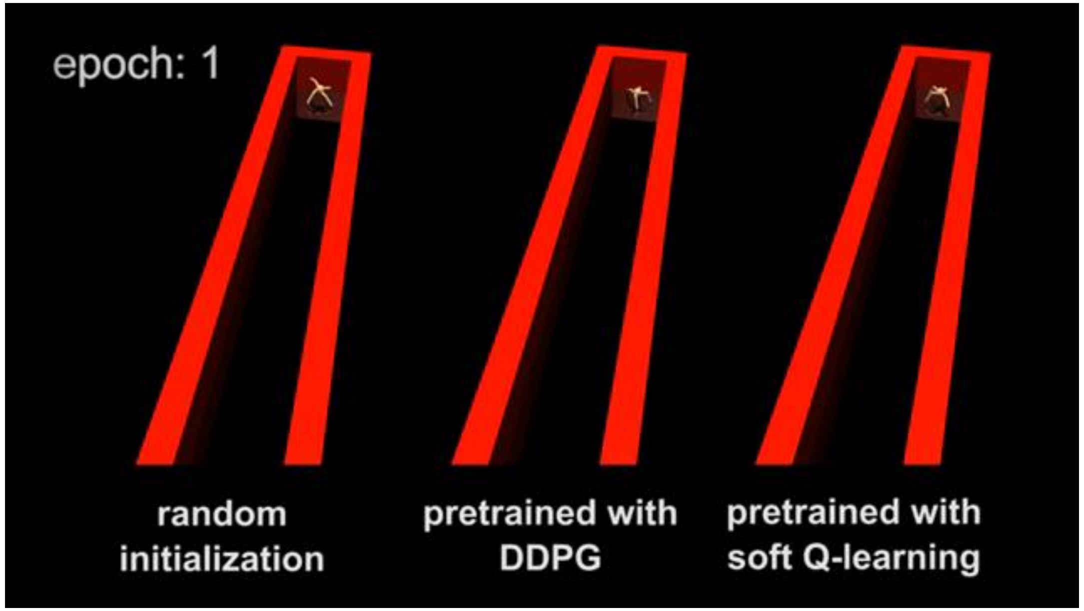


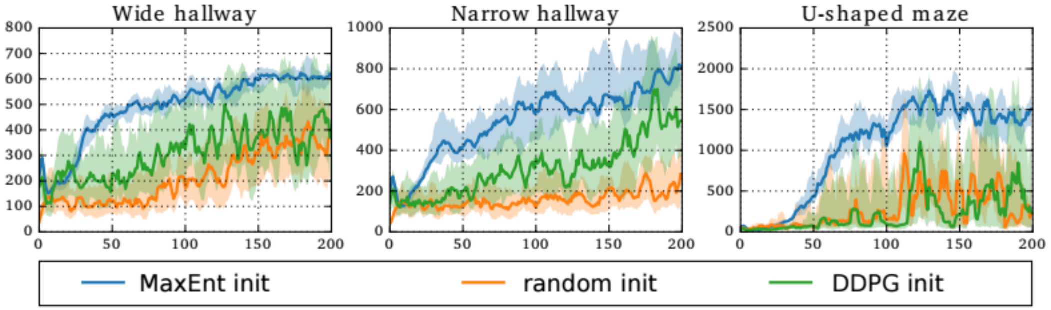
Trained with amortized SVGD to match $\pi(\mathbf{a}|\mathbf{s}) \propto \exp(Q(\mathbf{s},\mathbf{a}))$

Wang & Liu, '17

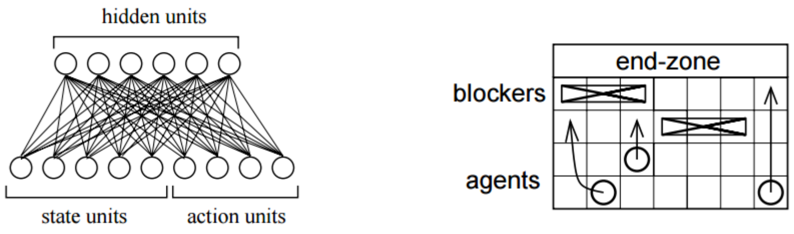
Stochastic energy-based policies provide pretraining





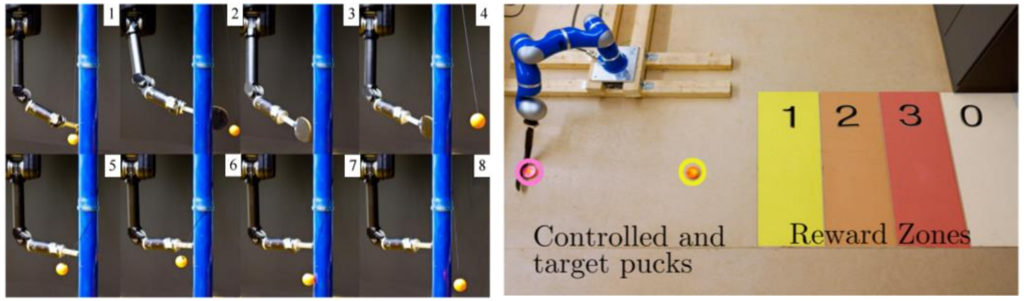


More work on maximum entropy policies

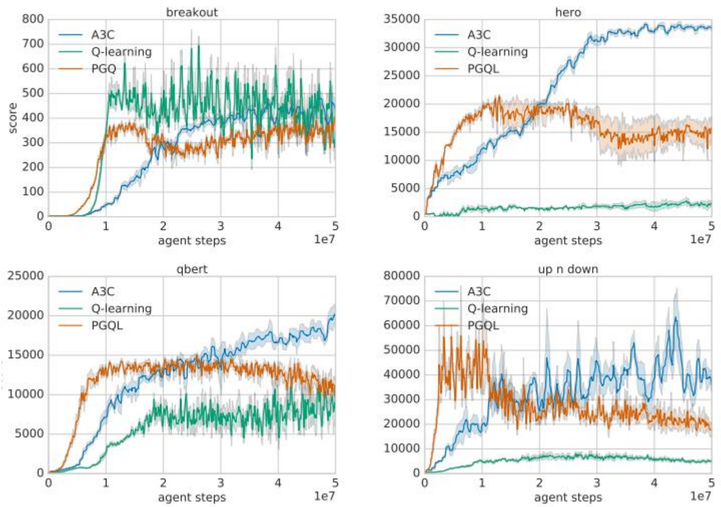


Sallans & Hinton. Using Free Energies to Represent Q-values in a Multiagent Reinforcement Learning Task. 2000.

Nachum et al. Bridging the Gap Between Value and Policy Based Reinforcement Learning. 2017.



Peters et al. Relative Entropy Policy Search. 2010.



O’Donoghue et al. Combining Policy Gradient and Q-Learning. 2017

Soft optimality suggested readings

- Todorov. (2006). Linearly solvable Markov decision problems: one framework for reasoning about soft optimality.
- Todorov. (2008). General duality between optimal control and estimation: primer on the equivalence between inference and control.
- Kappen. (2009). Optimal control as a graphical model inference problem: frames control as an inference problem in a graphical model.
- Ziebart. (2010). Modeling interaction via the principle of maximal causal entropy: connection between soft optimality and maximum entropy modeling.
- Rawlik, Toussaint, Vijaykumar. (2013). On stochastic optimal control and reinforcement learning by approximate inference: temporal difference style algorithm with soft optimality.
- Haarnoja*, Tang*, Abbeel, L. (2017). Reinforcement learning with deep energy based models: soft Q-learning algorithm, deep RL with continuous actions and soft optimality
- Nachum, Norouzi, Xu, Schuurmans. (2017). Bridging the gap between value and policy based reinforcement learning.
- Schulman, Abbeel, Chen. (2017). Equivalence between policy gradients and soft Q-learning.