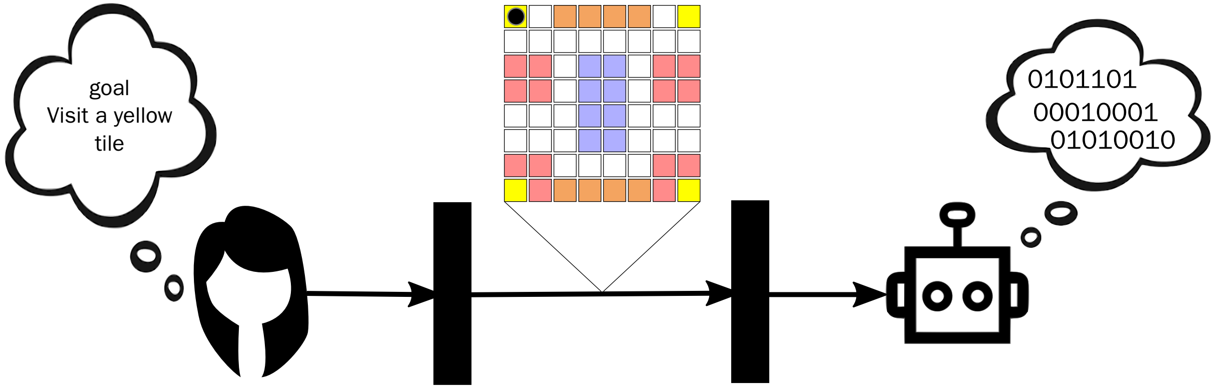
Inferring Specifications From Demonstrations

A Maximum (Causal) Entropy Approach

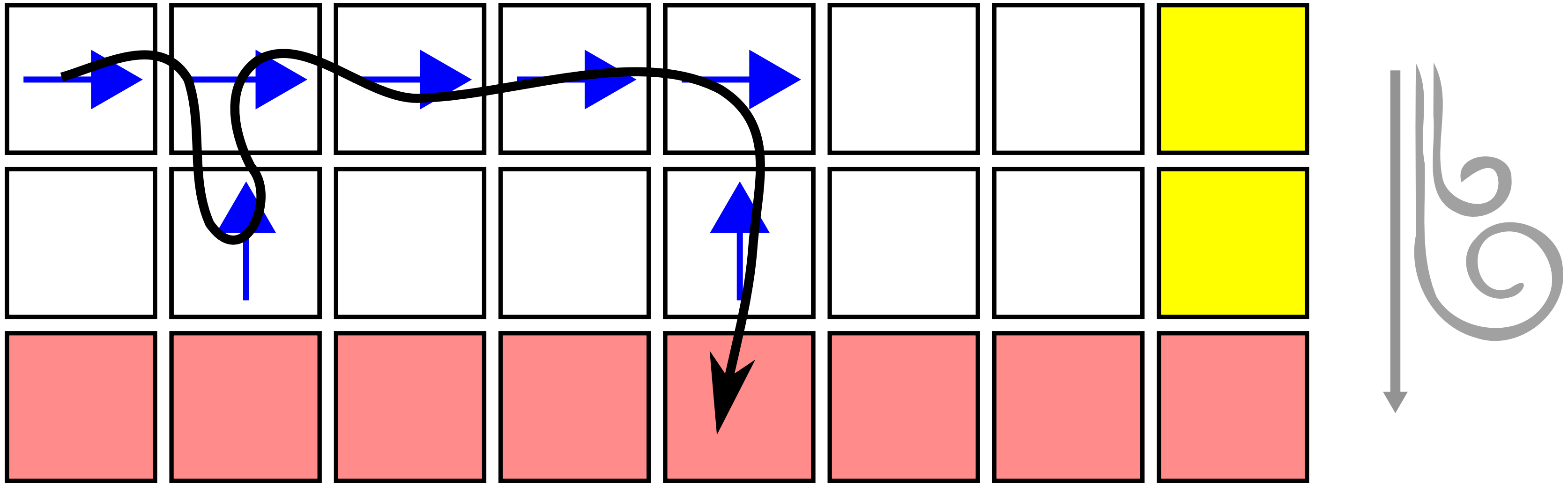


Marcell J. Vazquez-Chanlatte

Slides @ mjvc.me/simonsSP21

Collaborations with: Susmit Jha, Ashish Tiwari, Mark K. Ho, and Sanjit A. Seshia.

Motivating Example

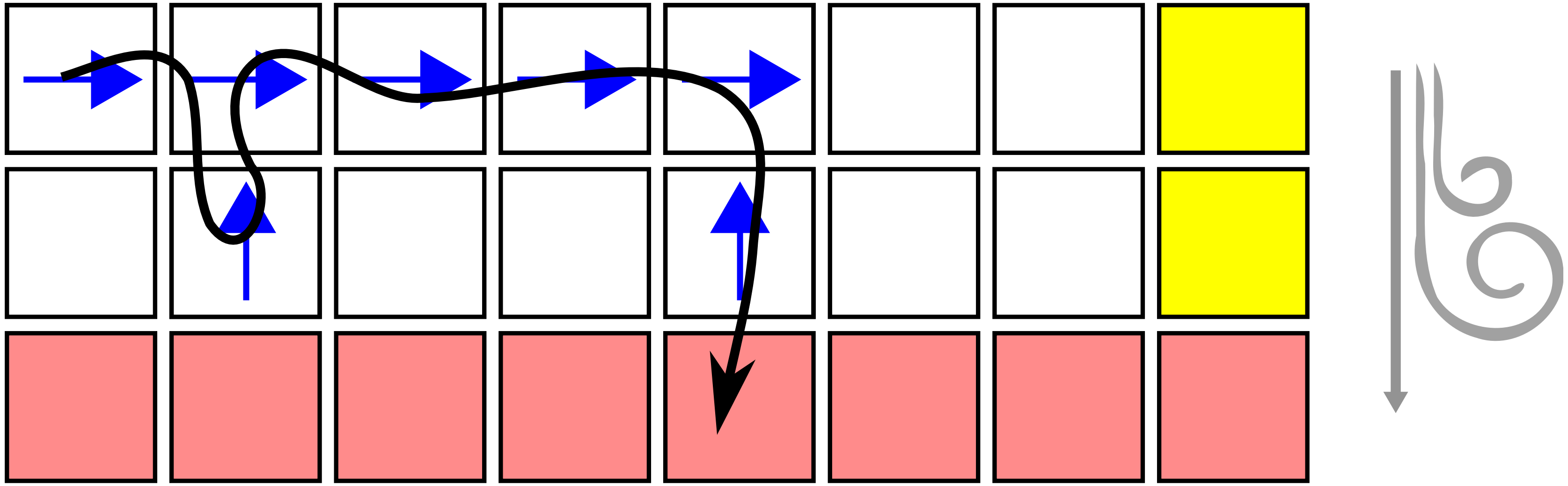


Consider an agent acting in the following stochastic grid world.

1. Set of actions: $\{\uparrow, \downarrow, \leftarrow, \rightarrow\}$.
2. $p = \frac{1}{32}$, slip and move $\downarrow$.

Q: What was the agent trying to do?

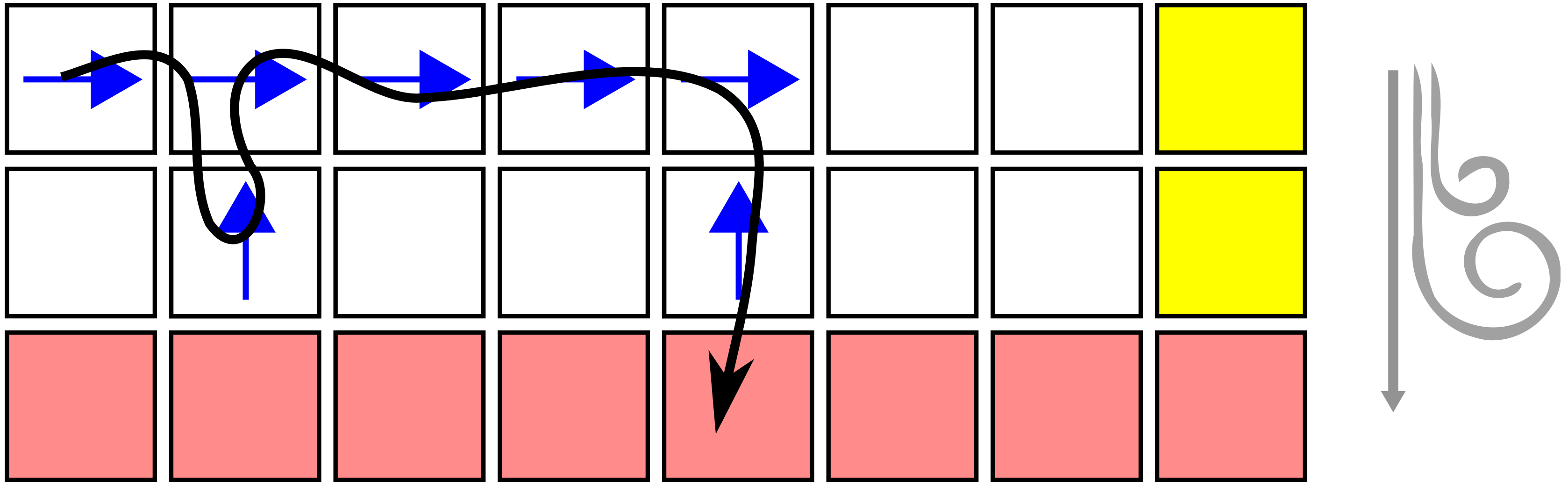
What was the agent trying to do?



Consider an agent acting in the following stochastic grid world.

Q: Did the agent intend to touch the **red** tile?

What was the agent trying to do?

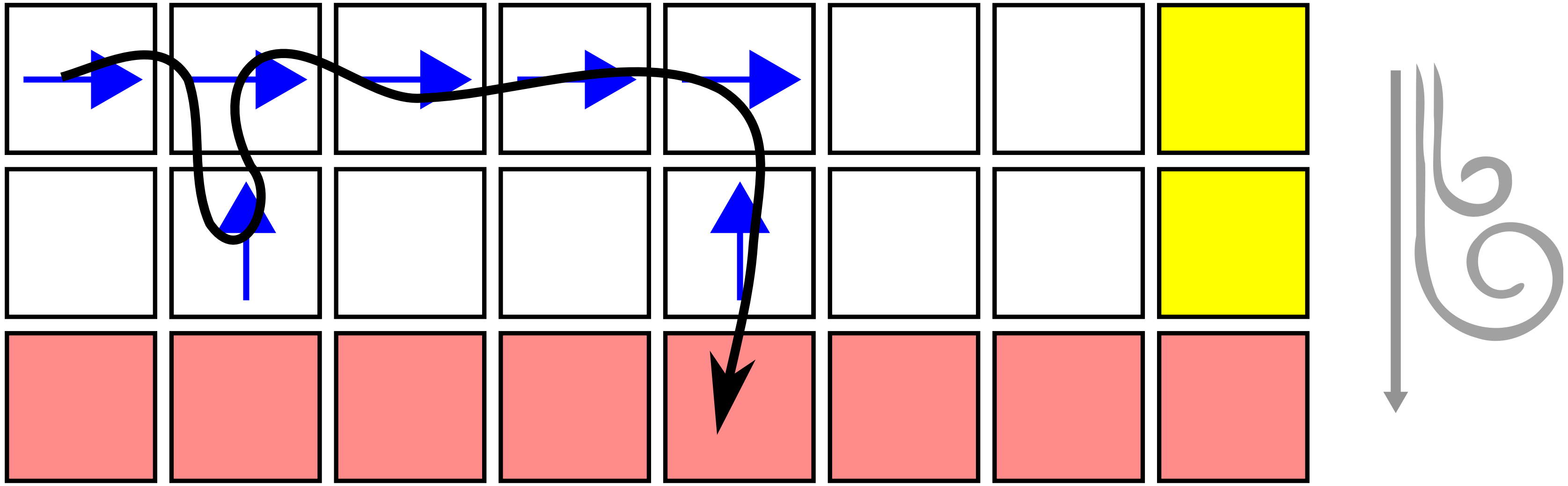


Consider an agent acting in the following stochastic grid world.

Q: Did the agent intend to touch the **red** tile? **A:** Probably Not.

Q: Did the agent intend to eventually touch a **yellow** tile?

What was the agent trying to do?

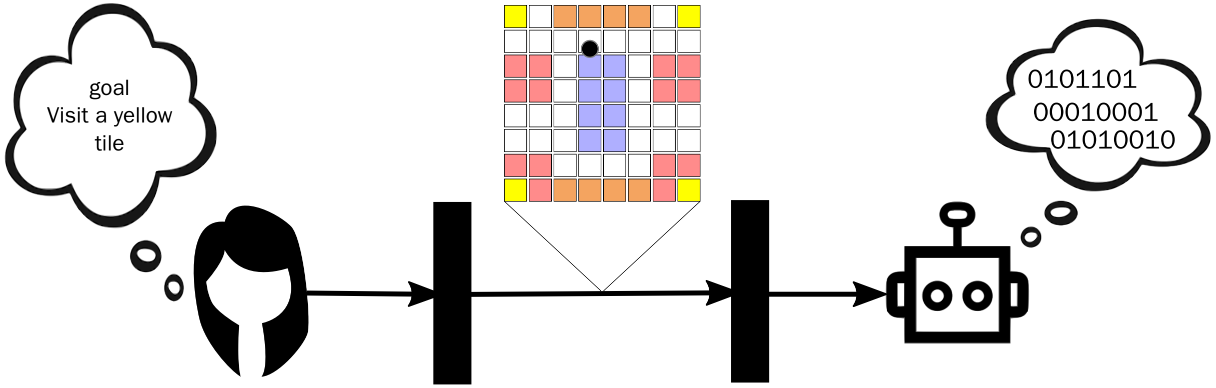


Consider an agent acting in the following stochastic grid world.

Q: Did the agent intend to touch the **red** tile? **A:** Probably Not.

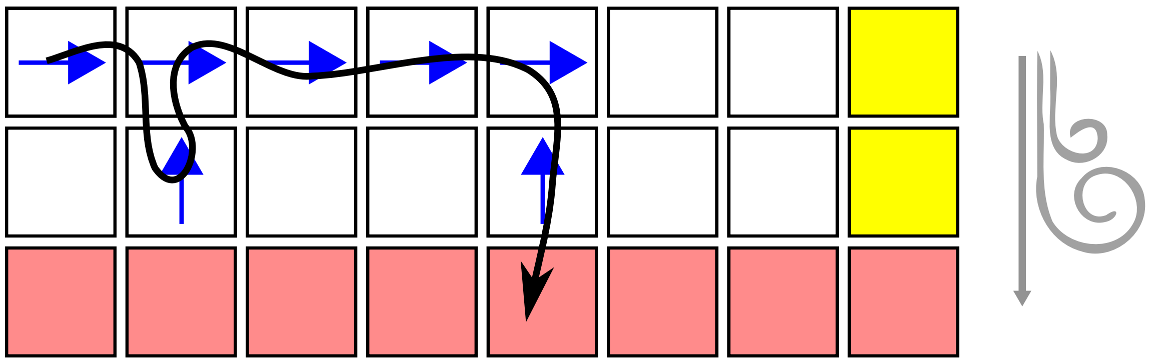
Q: Did the agent intend to eventually touch a **yellow** tile? **A:** Probably.

Communication through demonstrations

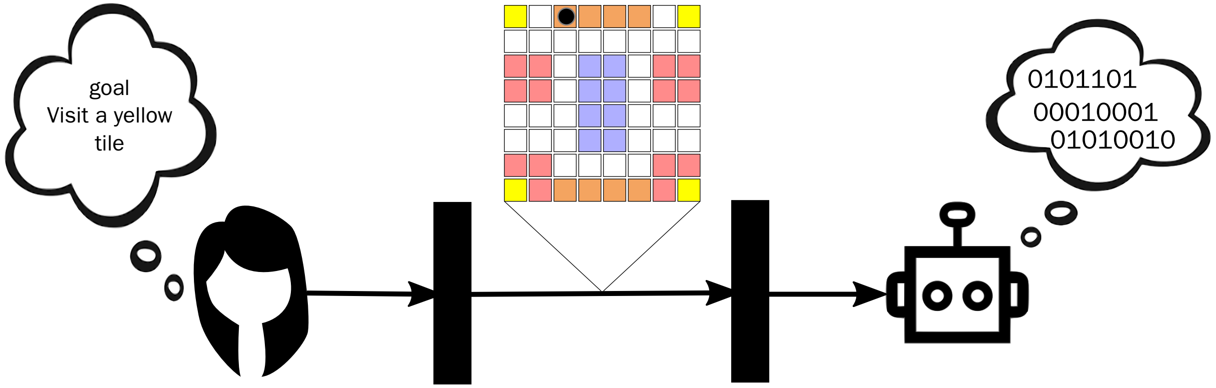


Demonstration information channel.

Can often learn given **unlabeled** demonstration errors!



Communication through demonstrations



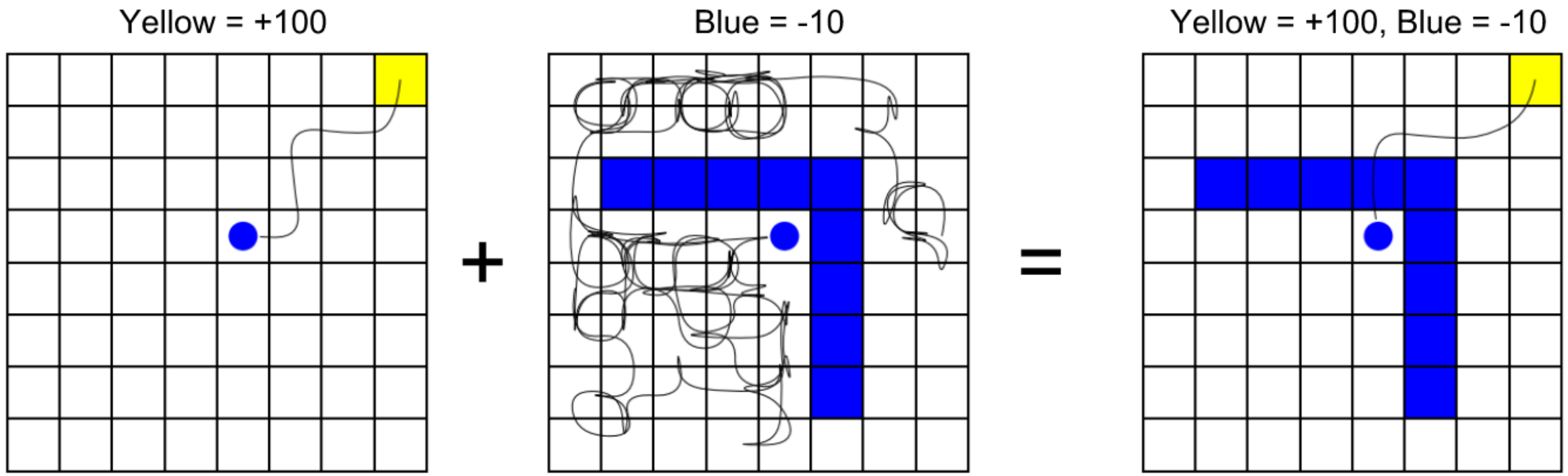
Demonstration information channel.

Goal: Develop algorithms to learn specifications from unlabeled demonstrations.

Q: Why not learn rewards?

Problems with rewards

Problem 1: Requires a “common currency” for reward.

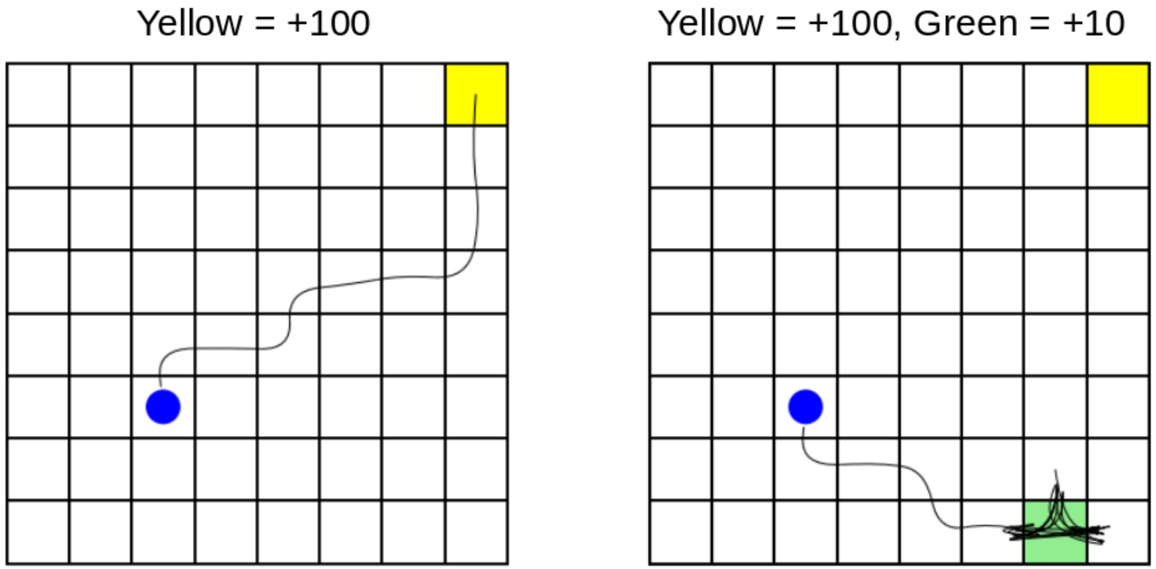


Littman, Topcu, Fu, Isbell, Wen & MacGlashan (2017)

How to safely compose in a dynamics invariant way?

Problems with rewards

Problem 2: Quantitative reward functions are usually Markov.

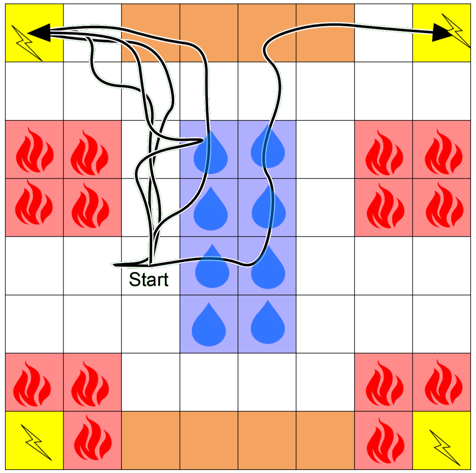


1. Dynamic States != Reward States
2. Beware the curse of history (Pineau et al 2003).
   Adding history can result in exponential state space explosion.

Specifications admit composition

Example Task

$$\varphi = \varphi_1 \wedge \varphi_2 \wedge \varphi_3$$



Example Gridworld Domain.

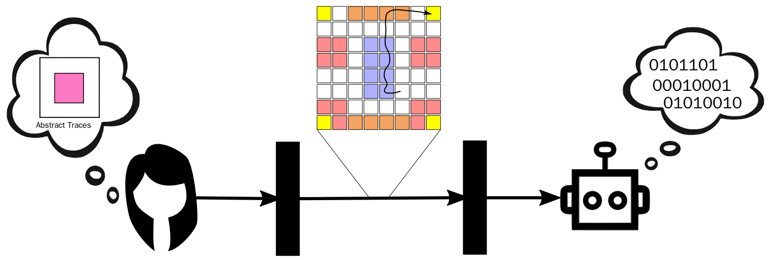
φ_1 = Eventually recharge.

φ_2 = Avoid lava.
φ_3 = If agent enters water, the agent must dry off before recharging.

Can learn incrementally or in parallel and then recompose.

Structure of the talk

Prelude - Problem Setup



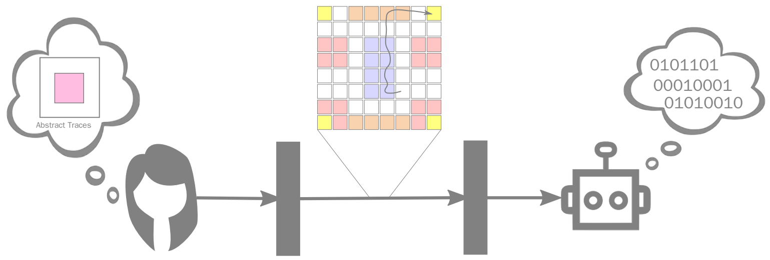
Act 1 - Naïve Problem Formulation

Act 2 - Exploiting Boolean Structure

Finale - Experiment

Structure of the talk

Prelude - Problem Setup



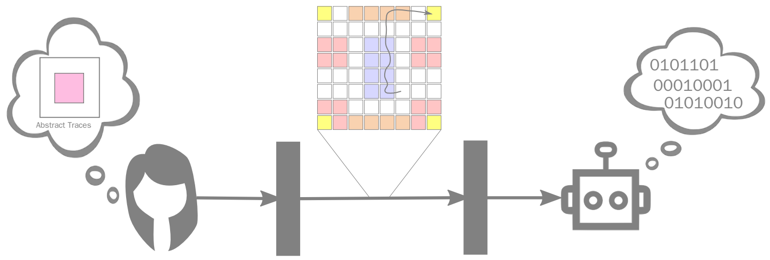
Act 1 - Naïve Problem Formulation

Act 2 - Exploiting Boolean Structure

Finale - Experiment

Structure of the talk

Prelude - Problem Setup



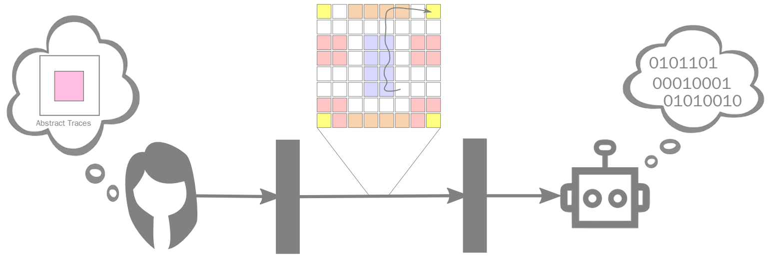
Act 1 - Naïve Problem Formulation

Act 2 - Exploiting Boolean Structure

Finale - Experiment

Structure of the talk

Prelude - Problem Setup



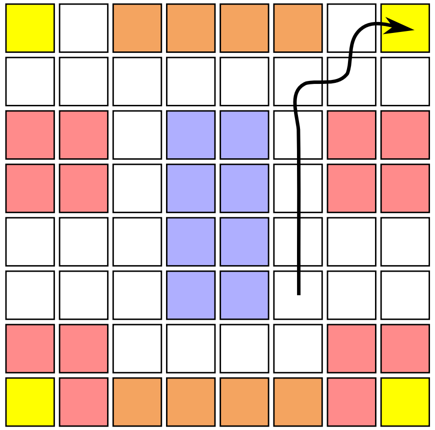
Act 1 - Naïve Problem Formulation

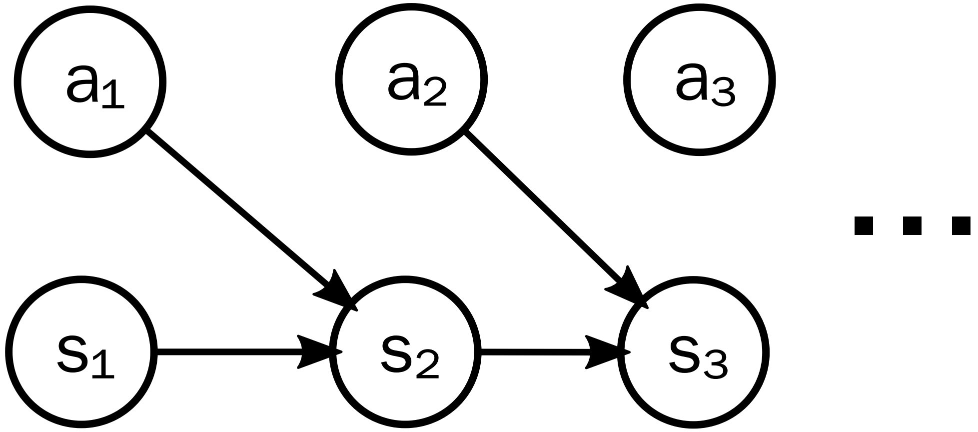
Act 2 - Exploiting Boolean Structure

Finale - Experiment

Basic definitions

1. Assume some fixed sets of **states** and **actions**.



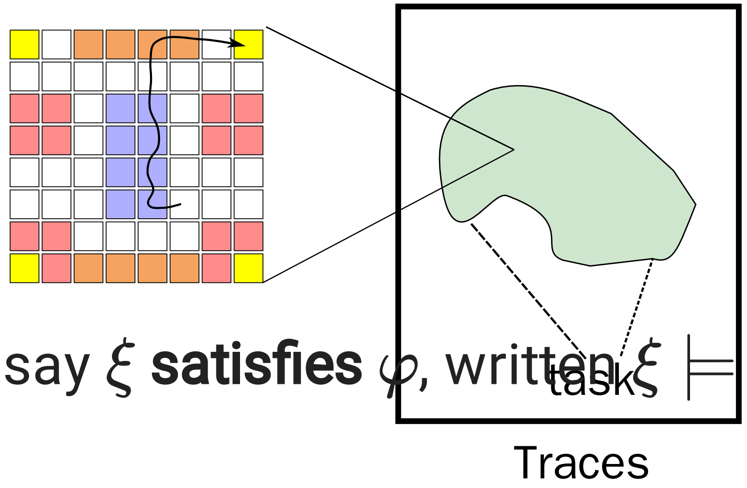


2. A **trace**, ξ, is a sequence of states and actions.

3. Assume all traces the same length, $\tau \in \mathbb{N}$.

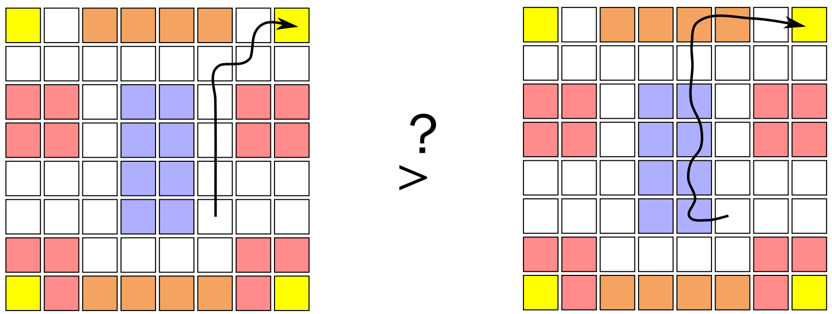
Basic definitions

1. Assume some fixed sets of **states** and **actions**.
2. A **trace**, ξ, is a sequence of states and actions.
3. Assume all traces the same length, $\tau \in \mathbb{N}$.
4. A (Boolean) **specification** φ, is a set of traces.



5. We say ξ **satisfies** φ, written $\xi \models \varphi$, if $\xi \in \varphi$.

No a-priori order on traces

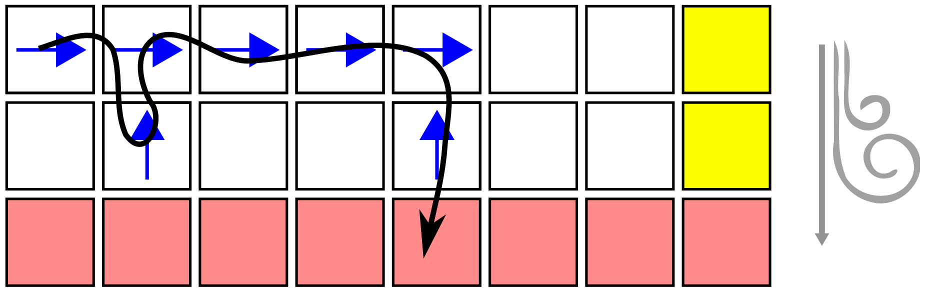


Agent model induces ordering.

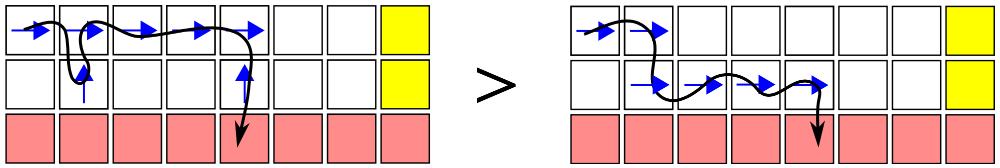
1. Need to know what moves are "risky".
2. Need to know agent's objective and competency.

Agent model induces ordering

- A **demonstration** of a task φ is an unlabeled example where the agent **tries** to satisfy φ.

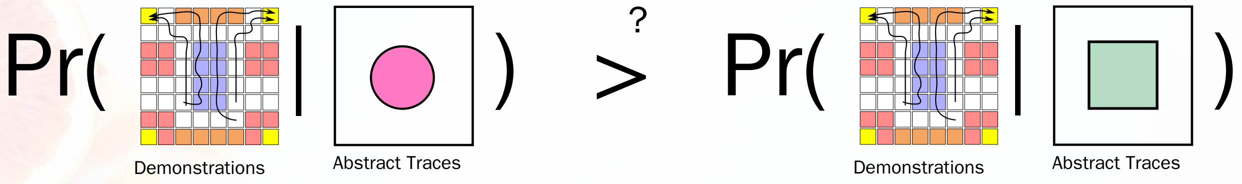


- Agency is key. Need a notion of **action**.
- Success probabilities induce an ordering.

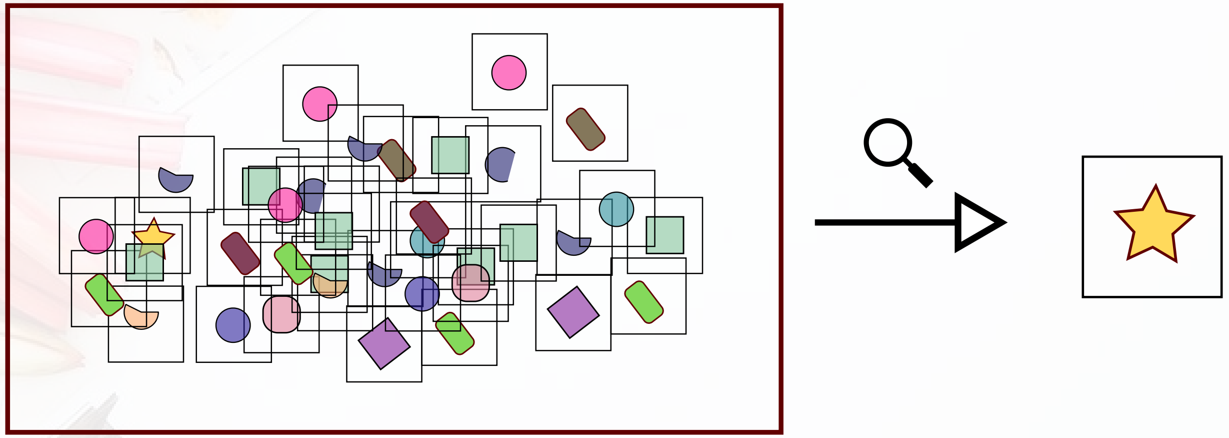


Solution Ingredients

1. Compare Likelihoods.



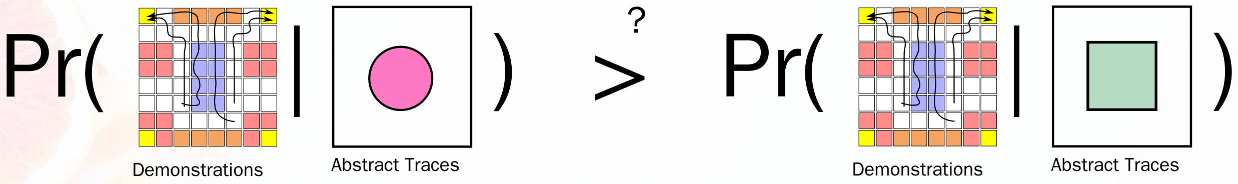
2. Search for likely specifications.



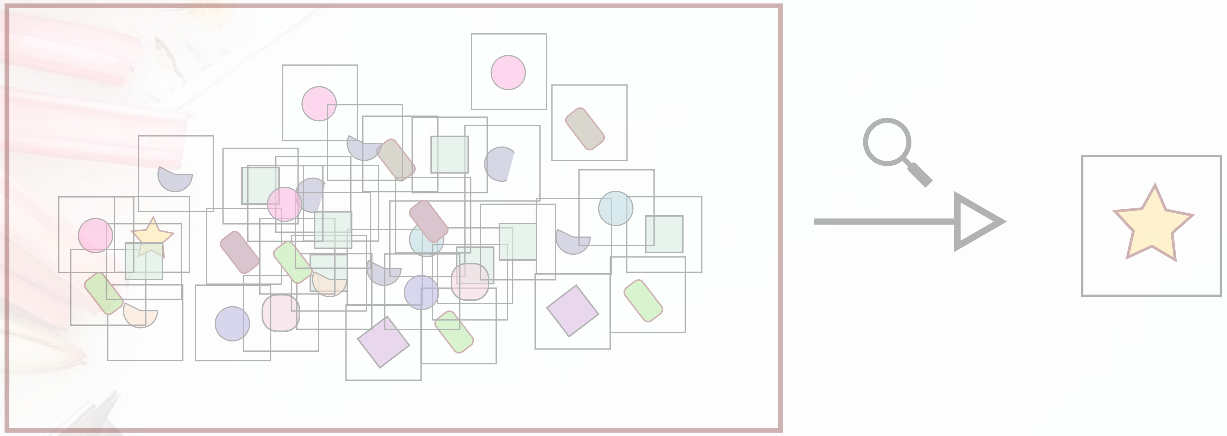
Concept Class

Solution Ingredients

1. Compare Likelihoods. **Focus on this today.**



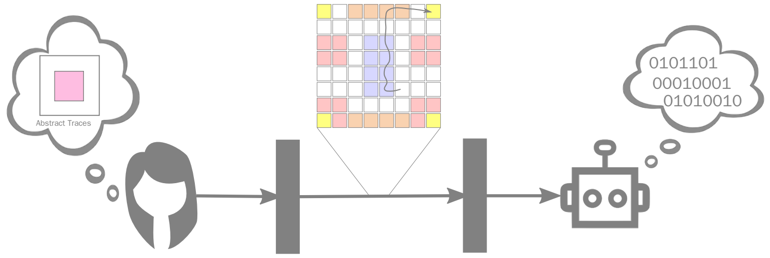
2. Search for likely specifications.



Concept Class

Structure of the talk

Prelude - Problem Setup



Act 1 - Naïve Problem Formulation

Act 2 - Exploiting Boolean Structure

Finale - Experiment

Structure of the talk

Prelude - Problem Setup

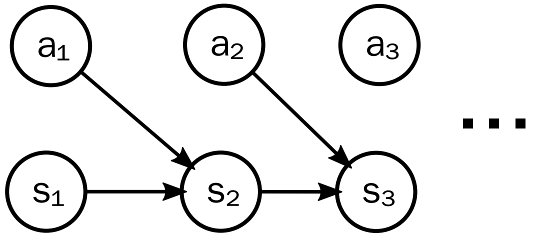
Act 1 - Naïve Problem Formulation

1. Cast problem as inverse reinforcement learning.
2. Apply principle of maximum causal entropy.

Act 2 - Exploiting Boolean Structure

Finale - Experiment

Inverse Reinforcement Learning



Assume agent is acting in a Markov Decision process and optimizing the sum of an unknown state reward, $r(s)$, i.e,:

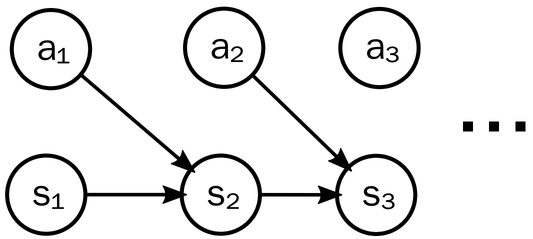
$$\max_{\pi}\left(\mathbb{E}_{s_{1:\tau}}\left(\sum_{i=1}^{\tau} r(s_i) \mid \pi\right)\right)$$

where

$$\pi(a \mid s) = \Pr(a \mid s)$$

Given a series of demonstrations, what reward, $r(s)$, best explains the behavior? (Abbeel and Ng 2004)

Inverse Reinforcement Learning



Given a series of demonstrations, what reward, $r(s)$, best explains the behavior? (Abbeel and Ng 2004)

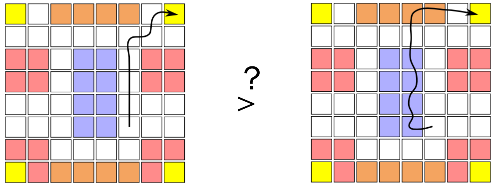
1. **Problem:** There is no unique solution as posed!

$$\Pr(r \mid \xi) = ?$$

2. **Idea:** Disambiguate via the **Principle of Maximum Causal Entropy.** (Ziebart, et al. 2010)

Idea: Reduce Specification Inference to IRL.

Q: What should the reward be?



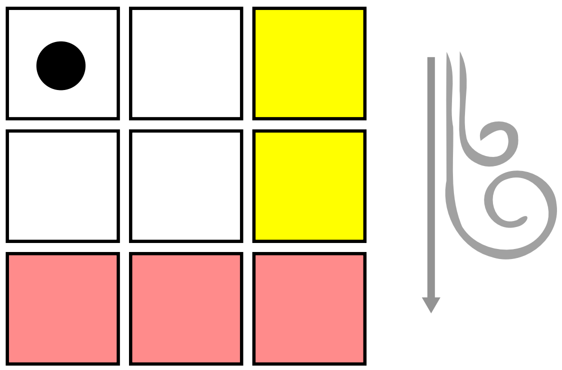
Proposal: Use indicator.

$$r(\xi) \triangleq \begin{cases} 1 & \text{if } \xi \in \varphi \\ 0 & \text{otherwise} \end{cases}$$

Idea: Reduce Specification Inference to IRL.

$$r(\xi) \triangleq \begin{cases} 1 & \text{if } \xi \in \varphi \\ 0 & \text{otherwise} \end{cases}$$

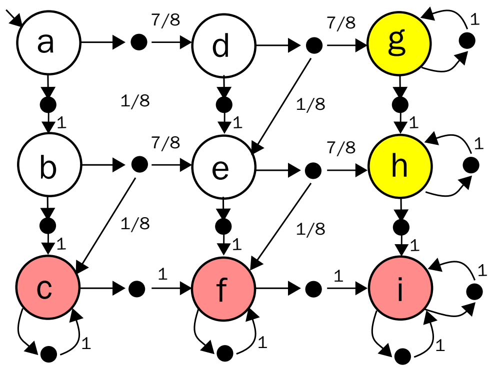
Note: States are now traces.



Idea: Reduce Specification Inference to IRL.

$$r(\xi) \triangleq \begin{cases} 1 & \text{if } \xi \in \varphi \\ 0 & \text{otherwise} \end{cases}$$

Note: States are now traces.

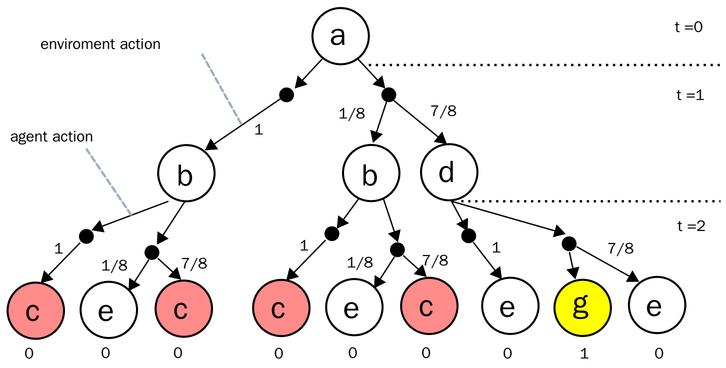


Suppose φ is over traces of length 2.

Idea: Reduce Specification Inference to IRL.

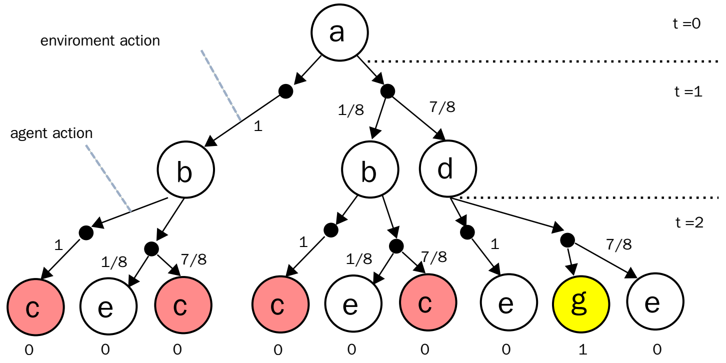
$$r(\xi) \triangleq \begin{cases} 1 & \text{if } \xi \in \varphi \\ 0 & \text{otherwise} \end{cases}$$

Note: States are now traces.



Suppose φ is over traces of length 2.

Idea: Reduce Specification Inference to IRL.



Problem: Naïve reduction leds to exponential blow up.

Post-pone this concern for now.

Structure of the talk

Prelude - Problem Setup

Act 1 - Naïve Problem Formulation

1. Cast problem as inverse reinforcement learning.

$$r(\xi) \triangleq \begin{cases} 1 & \text{if } \xi \in \varphi \\ 0 & \text{otherwise} \end{cases}$$

2. Apply principle of maximum causal entropy.

Act 2 - Exploiting Boolean Structure

Finale - Experiment

Structure of the talk

Prelude - Problem Setup

Act 1 - Naïve Problem Formulation

1. Cast problem as inverse reinforcement learning.

$$r(\xi) \triangleq \begin{cases} 1 & \text{if } \xi \in \varphi \\ 0 & \text{otherwise} \end{cases}$$

2. Apply principle of maximum causal entropy.

Act 2 - Exploiting Boolean Structure

Finale - Experiment

High Entropy Policies are Robust

Key problem

Given φ, was is demonstrator likely to do?

$\Pr(A_t \mid S_{1:t})$ = ?

Note: Maximum causal entropy forecaster minimizes worst case prediction log-loss. (Ziebart, et al. 2010)

Maximum causal entropy $\rightarrow$ Robust agent proxy

Maximum Causal Entropy

$$\Pr(A_t \mid S_{1:t}) = \ ?$$

Key Idea: Don't commit more than the observations require.

Formally: Maximize expected causal entropy.

$$H(A_{1:\tau} \mid\mid S_{1:\tau}) = \sum_{t=1}^{T} H\Big(A_t \mid S_{1:t}\Big)$$

subject to expected reward matching.

Maximum Causal Entropy

$$\Pr(A_t \mid S_{1:t}) = \ ?$$

Key Idea: Don't commit more than the observations require.

Formally: Maximize expected causal entropy.

$$H(A_{1:\tau} \,||\, S_{1:\tau}) = \sum_{t=1}^{T} H\Big(A_t \mid S_{1:t}\Big)$$

subject to $\mathbb{E}[r(S_{1:\tau})] = r^*.$

Maximum Causal Entropy

$$\Pr(A_t \mid S_{1:t}) = \ ?$$

Key Idea: Don't commit more than the observations require.

Formally: Maximize expected causal entropy.

$$H(A_{1:\tau} \mid\mid S_{1:\tau}) = \sum_{t=1}^{T} H\Big(A_t \mid S_{1:t}\Big)$$

subject to $\Pr(S_{1:\tau} \in \varphi) = p^*$.

Will consider two cases

a.

$$H(A_{1:T} \,||\, S_{1:T}) \approx H(A_{1:T} \mid S_{1:T})$$

"Learning Task Specifications from Demonstrations." NeurIPS 2018

b.

$$H(A_{1:T} \,||\, S_{1:T}) \not\approx H(A_{1:T} \mid S_{1:T})$$

"Maximum Causal Entropy Specification Inference from Demonstrations.", CAV 2020

Lets start with MaxEnt case

a.

$$H(A_{1:T} \,||\, S_{1:T}) \approx H(A_{1:T} \mid S_{1:T})$$

"Learning Task Specifications from Demonstrations." NeurIPS 2018

b.

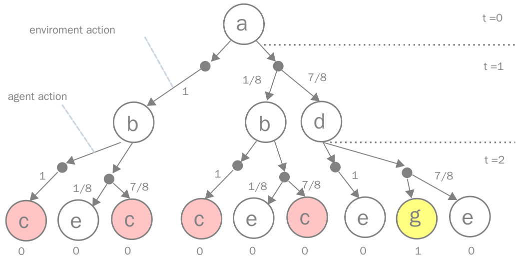
$$H(A_{1:T} \,||\, S_{1:T}) \not\approx H(A_{1:T} \mid S_{1:T})$$

"Maximum Causal Entropy Specification Inference from Demonstrations.", CAV 2020

Structure of the talk

Prelude - Problem Setup

Act 1 - Naïve Problem Formulation

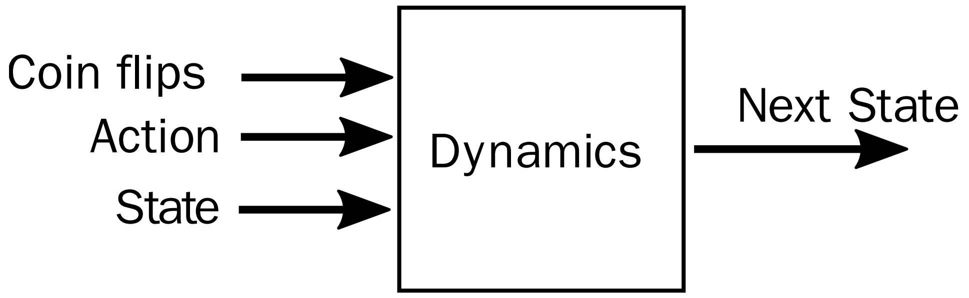


Act 2 - Exploiting Boolean Structure

Finale - Experiment

Change of perspective

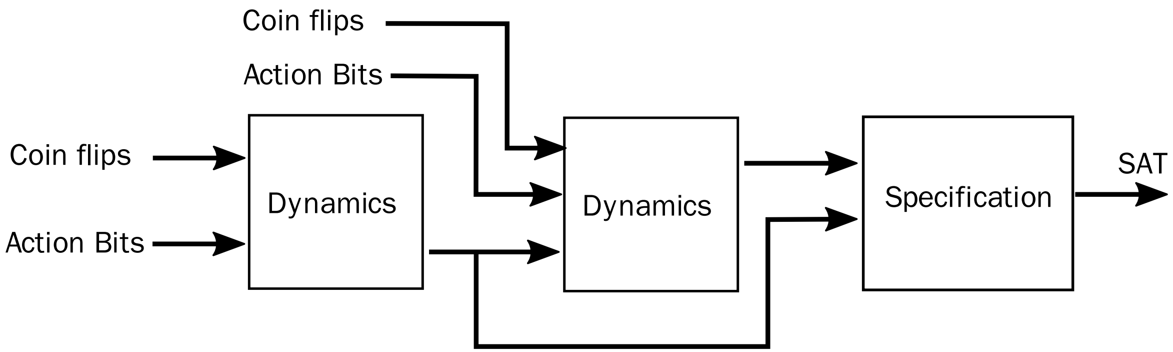
Random bit model: Represent Markov Decision Process as deterministic transition system with access to n_c coin flips.



$$\text{Dynamics} : S \times \{0,1\}^{n_a+n_c} \to S$$

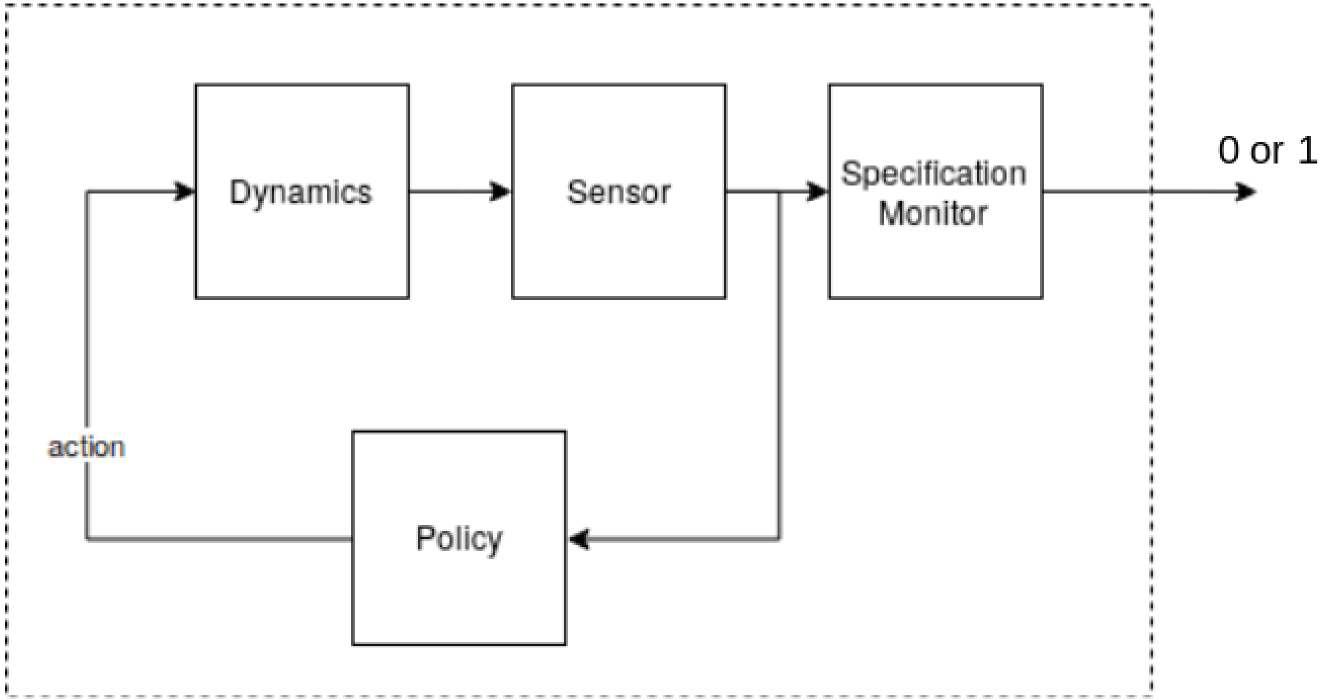
Change of perspective

Unrolling and composing with specification results in a predicate.



$$\psi : \{0,1\}^{\tau \cdot (n_a + n_c)} \to \{0,1\}$$

Policy closes the loop

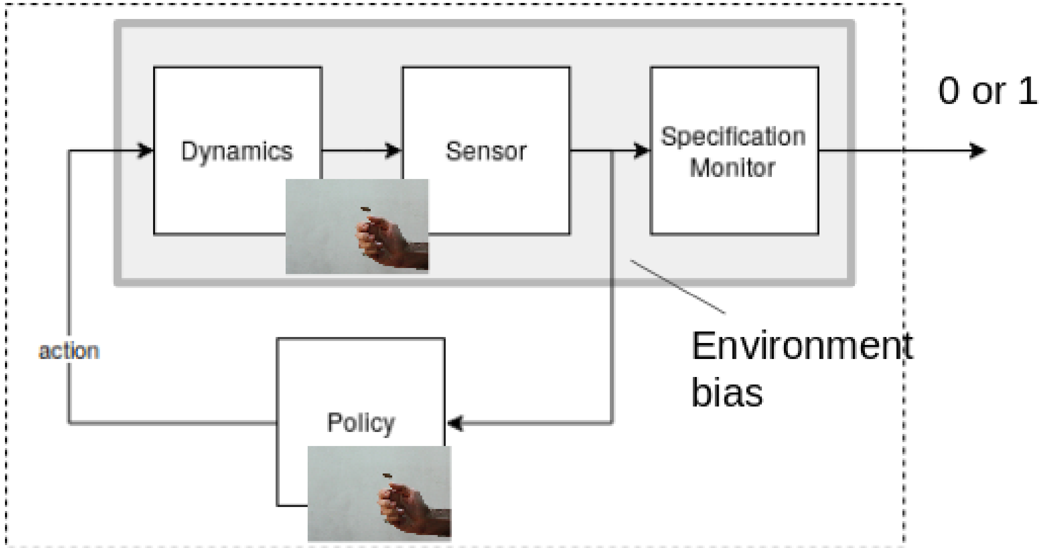


Looks like a biased coin

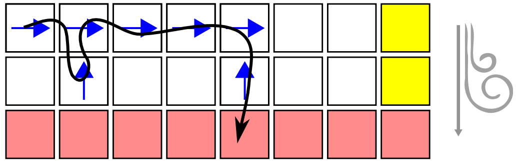
Observe satisfaction probability, p_{φ}.

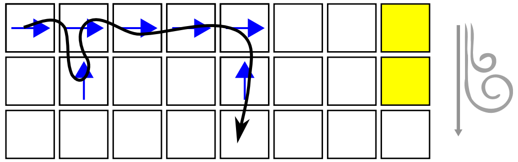
Need to be consistent with Bernoulli random variable.

Pulling back the curtain

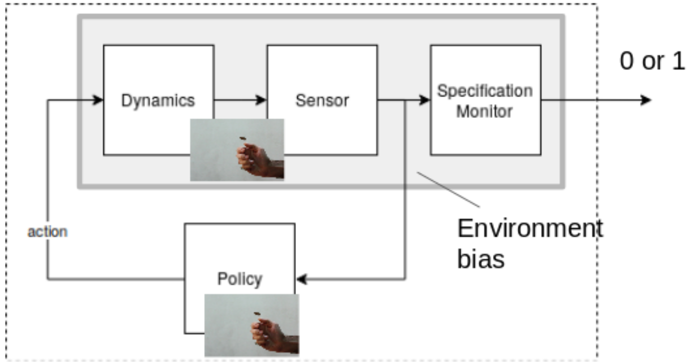


Satisfaction probability, p_φ, affected by policy and how "easy" the specification/dynamics combination is.





Policy doesn't need to be reactive

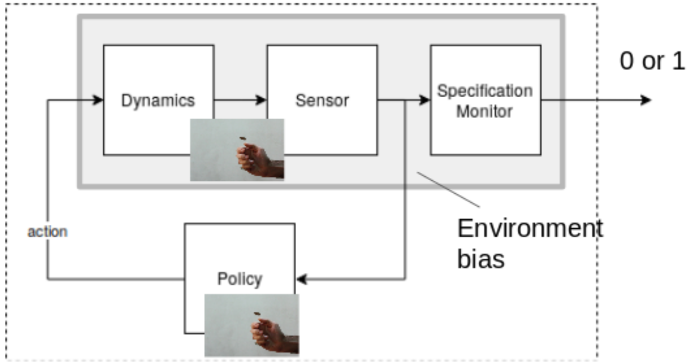


$$H(A_{1:\tau} \,||\, S_{1:\tau}) \approx H(A_{1:\tau} \mid S_{1:\tau})$$

"Learning Task Specifications from Demonstrations." NeurIPS 2018

Effects separable in MaxEnt case

Effects separable in MaxEnt case



$$p_\varphi \triangleq \Pr(\xi \models \varphi \mid \text{teacher } \pi) \qquad q_\varphi \triangleq \Pr(\xi \models \varphi \mid \text{uniform } A_{1:\tau})$$

1. The Maximum Entropy Distribution given p_φ is:

$$\Pr(S_{1:\tau} \mid \text{demos}, \varphi) \propto \begin{cases} \frac{p_\varphi}{q_\varphi} & \text{if } S_{1:\tau} \in \varphi \\ \frac{p_{\neg\varphi}}{q_{\neg\varphi}} & \text{if } S_{1:\tau} \notin \varphi \end{cases}$$

2. **Note:** When the dynamics are deterministic, this recovers the **size principle** from **concept learning**! (Tenenbaum 1999)

Maximum Entropy Likelihood given i.i.d. demos

Additional Assumptions

- Teacher at least as good as random: $p_\varphi \geq q_\varphi$
- Demonstrations, demos given i.i.d.
- Demonstrations are representative: $n \cdot p_\varphi \approx \#\{\xi_i \in \varphi\}$.
- $P_\varphi \triangleq \text{coin with bias } p_\varphi \quad Q_\varphi \triangleq \text{coin with bias } q_\varphi$

...

$$\Pr(\text{demos} \mid \varphi) \propto \underbrace{1[p_\varphi \geq q_\varphi]}_{\text{better than random}} \exp\left(n \cdot \underbrace{D_{KL}(P_\varphi \,||\, Q_\varphi)}_{\text{InfoGain over random actions.}}\right)$$

Aside: Can be interpreted as quantifying the atypicality of demos over random action hypothesis. (Sanov's Theorem 1957)

Max Entropy and Max Causal Entropy

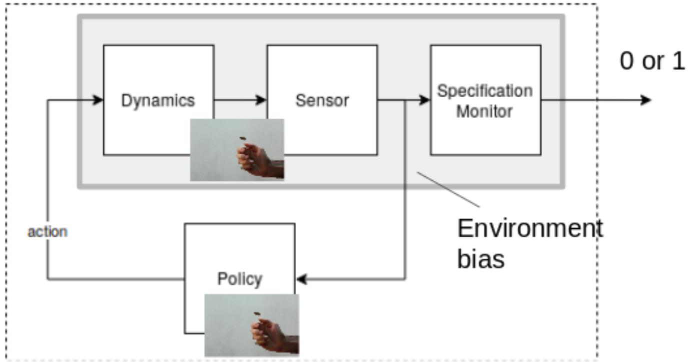
a. $$H(A_{1:\tau} \,||\, S_{1:\tau}) \approx H(A_{1:\tau} \mid S_{1:\tau})$$

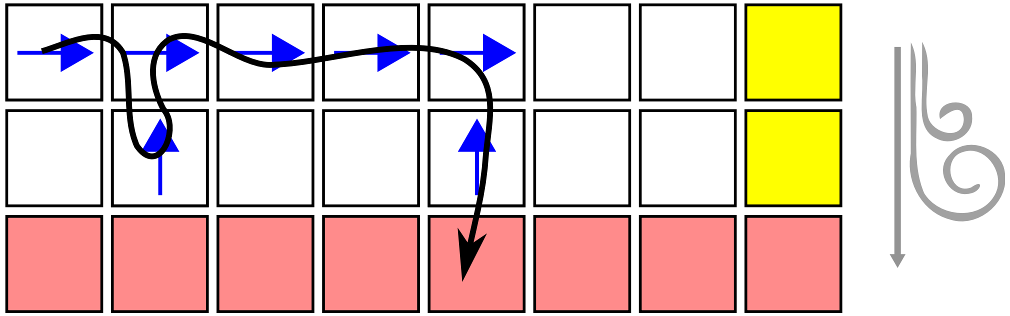
"Learning Task Specifications from Demonstrations." NeurIPS 2018

b. $$H(A_{1:\tau} \,||\, S_{1:\tau}) \not\approx H(A_{1:\tau} \mid S_{1:\tau})$$

"Maximum Causal Entropy Specification Inference from Demonstrations.", CAV 2020

Generally need to be reactive.





$$H(A_{1:T} \,||\, S_{1:T}) \not\approx H(A_{1:T} \mid S_{1:T})$$

"Maximum Causal Entropy Specification Inference from Demonstrations.", CAV 2020

Soft Bellman backup

Maximum Causal Entropy Policy

$$\log\left(\pi_\theta(a_{1:t} \mid s_{1:t})\right) = Q_\theta(a_{1:t}, s_{1:t}) - V_\theta(s_{1:t})$$

where

$$V_\theta(s_{1:t}) \triangleq \begin{cases} \ln \sum_{a_{1:t}} e^{Q_\theta(a_{1:t}, s_{1:t})} & \text{if } t \neq \tau, \\ \theta \cdot [s_{1:\tau} \in \varphi] & \text{otherwise.} \end{cases}$$

$$Q_\theta(a_{1:t}, s_{1:t}) \triangleq \mathbb{E}_{s_{1:t+1}}\left[V_\theta(s_{t+1}) \mid s_{1:t}, a_{1:t}\right]$$

Find θ to match p^*.

Soft Bellman backup

Maximum Causal Entropy Policy

$$V_\theta(s_{1:t}) \triangleq \begin{cases} \ln \sum_{a_{1:t}} e^{Q_\theta(a_{1:t}, s_{1:t})} & \text{if } t \neq \tau, \\ \theta \cdot [s_{1:\tau} \in \varphi] & \text{otherwise.} \end{cases}$$

$$Q_\theta(a_{1:t}, s_{1:t}) \triangleq \mathbb{E}_{s_{1:t+1}} \left[V_\theta(s_{t+1}) \mid s_{1:t}, a_{1:t} \right]$$

Focus on recursive soft-value calculation.

Looks like standard Bellman backup

Maximum Causal Entropy Policy

$$V_\theta(s_{1:t}) \triangleq \begin{cases} \text{smax}_{a_{1:t}} Q_\theta(a_{1:t}, s_{1:t}) & \text{if } t \neq \tau, \\ \theta \cdot 1[s_{1:\tau} \in \varphi] & \text{otherwise.} \end{cases}$$

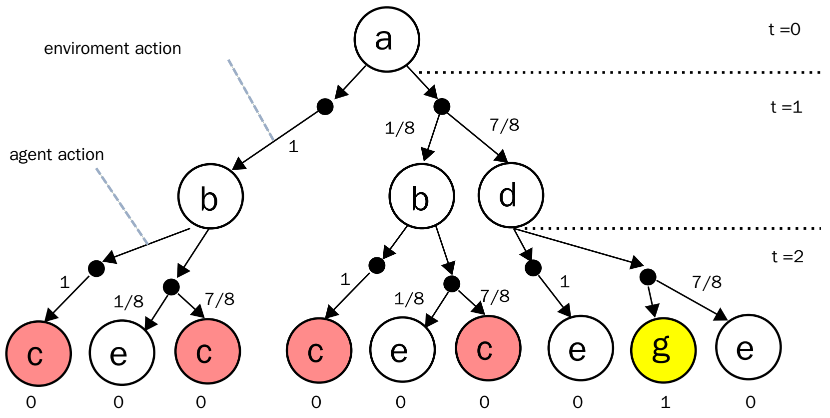
$$Q_\theta(a_{1:t}, s_{1:t}) \triangleq \mathbb{E}_{s_{1:t+1}} \left[V_\theta(s_{t+1}) \mid s_{1:t}, a_{1:t} \right]$$

max $\mapsto$ smooth maximum.

Soft Bellman backup

$$V_\theta(s_{1:t}) \triangleq \begin{cases} \operatorname{smax}_{a_{1:t}} Q_\theta(a_{1:t}, s_{1:t}) & \text{if } t \neq \tau, \\ \theta \cdot 1[s_{1:\tau} \in \varphi] & \text{otherwise.} \end{cases}$$

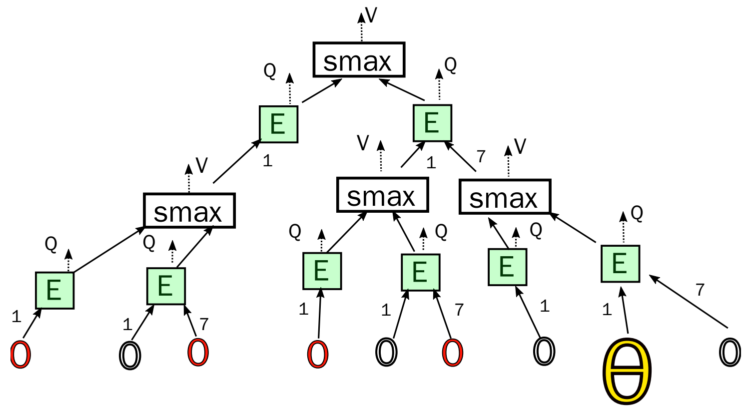
$$Q_\theta(a_{1:t}, s_{1:t}) \triangleq \mathbb{E}_{s_{1:t+1}}\left[V_\theta(s_{t+1}) \mid s_{1:t}, a_{1:t}\right]$$



Backup as computation graph

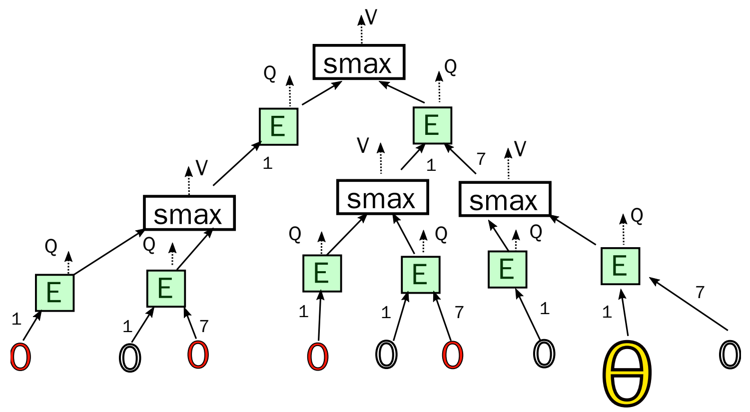
$$V_\theta(s_{1:t}) \triangleq \begin{cases} \text{smax}_{a_{1:t}} Q_\theta(a_{1:t}, s_{1:t}) & \text{if } t \neq \tau, \\ \theta \cdot 1[s_{1:\tau} \in \varphi] & \text{otherwise.} \end{cases}$$

$$Q_\theta(a_{1:t}, s_{1:t}) \triangleq \mathbb{E}_{s_{1:t+1}} \left[V_\theta(s_{t+1}) \mid s_{1:t}, a_{1:t} \right]$$



Find θ to match p^* .

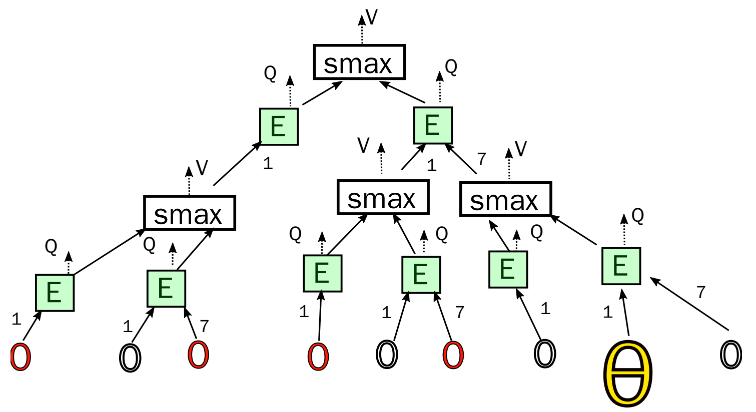
Backup as computation graph



Note: Satisfaction probability grows monotonically in θ.

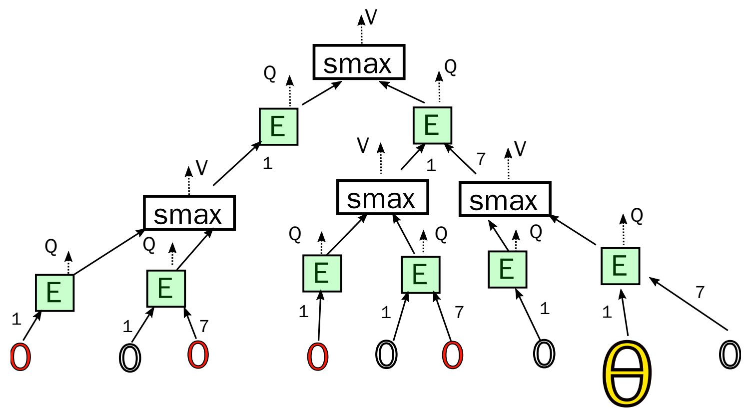
Can binary search for θ such that satisfaction probability matches data.

Backup as computation graph



Problem: Unrolled tree grows exponentially in horizon!

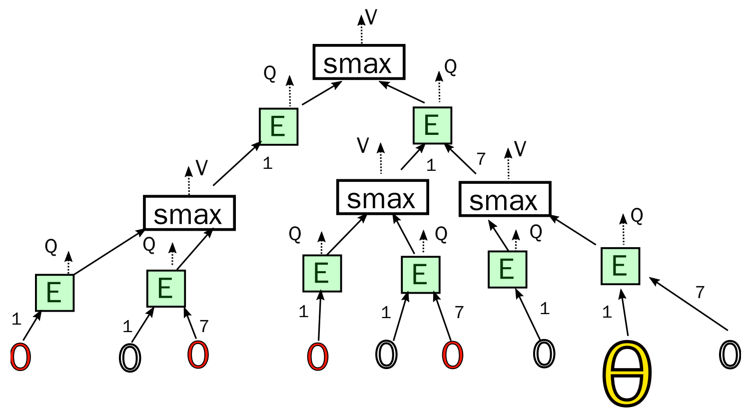
Backup as computation graph



Observation 1: A lot of shared structure in computation graph.

Observation 2: System and environment actions are ordered.

Backup as computation graph

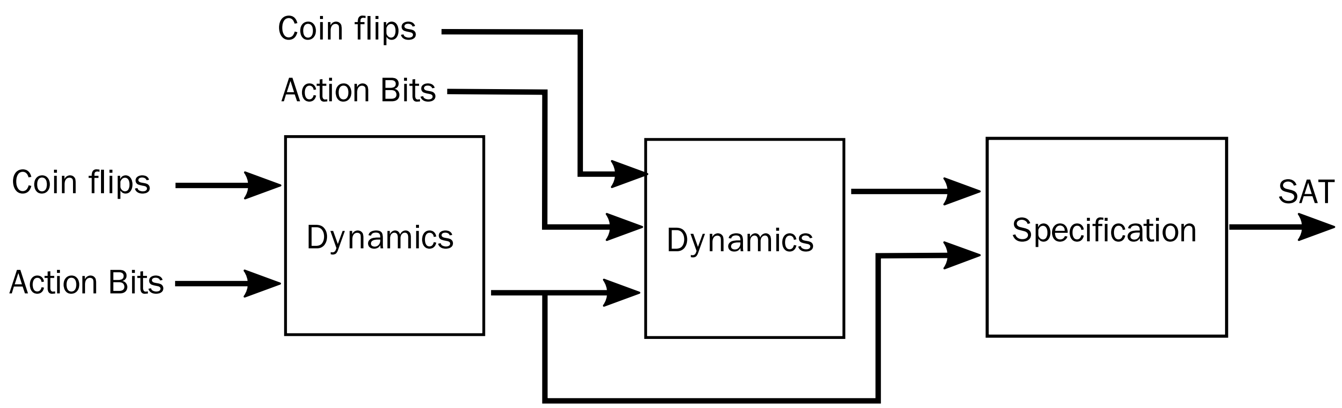


Idea: Encode graph as a binary predicate

$$\psi : \{0,1\}^n \to \{0,1\}$$

and represent as Reduced Ordered Binary Decision Diagram (Bryant 1986).

Random Bit Model



Idea: Encode graph as a binary predicate

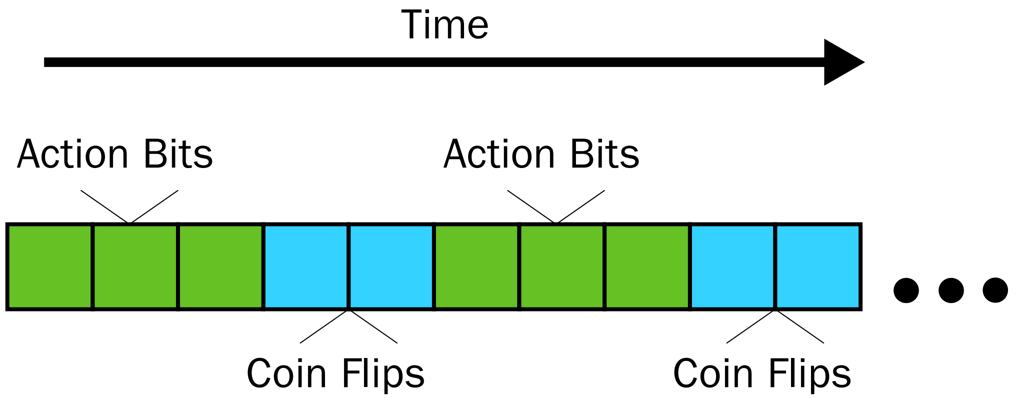
$$\psi : \{0,1\}^{\tau \cdot (n_a + n_c)} \to \{0,1\}$$

and represent as Reduced Ordered Binary Decision Diagram (Bryant 1986).

Random Bit Model

$$\psi : \{0,1\}^{\tau\cdot(n_a+n_c)} \to \{0,1\}$$

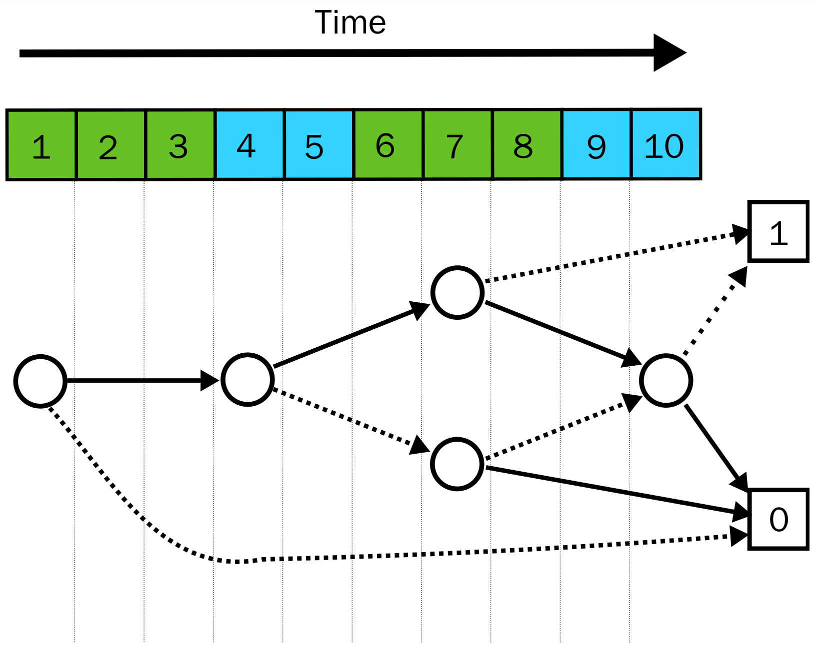
Proposal: Represent ψ as Binary Decision Diagram with bits in causal order.



Random Bit Model

$$\psi : \{0, 1\}^{\tau \cdot (n_a + n_c)} \to \{0, 1\}$$

Proposal: Represent ψ as Binary Decision Diagram with bits in causal order.



Maximum Causal Entropy and BDDs

Q: Can Maximum Entropy Causal Policy be computed on causally ordered BDDs? **A:** Yes!

1. Associativity of smax and $\mathbb{E}$.

$$\mathrm{smax}(\alpha_1, \ldots, \alpha_4) = \ln(\sum_{i=1}^{4} e^{\alpha_i})$$

Maximum Causal Entropy and BDDs

Q: Can Maximum Entropy Causal Policy be computed on causally ordered BDDs? **A:** Yes!

1. Associativity of smax and $\mathbb{E}$.

$$\mathrm{smax}(\alpha_1, \ldots, \alpha_4) = \ln(e^{\ln(e^{\alpha_1} + e^{\alpha_2})} + e^{\ln(e^{\alpha_3} + e^{\alpha_4})})$$

Maximum Causal Entropy and BDDs

Q: Can Maximum Entropy Causal Policy be computed on causally ordered BDDs? **A:** Yes!

1. Associativity of smax and $\mathbb{E}$.

$$\mathrm{smax}(\alpha_1, \ldots, \alpha_4) = \mathrm{smax}(\mathrm{smax}(\alpha_1, \alpha_2), \mathrm{smax}(\alpha_3, \alpha_4))$$

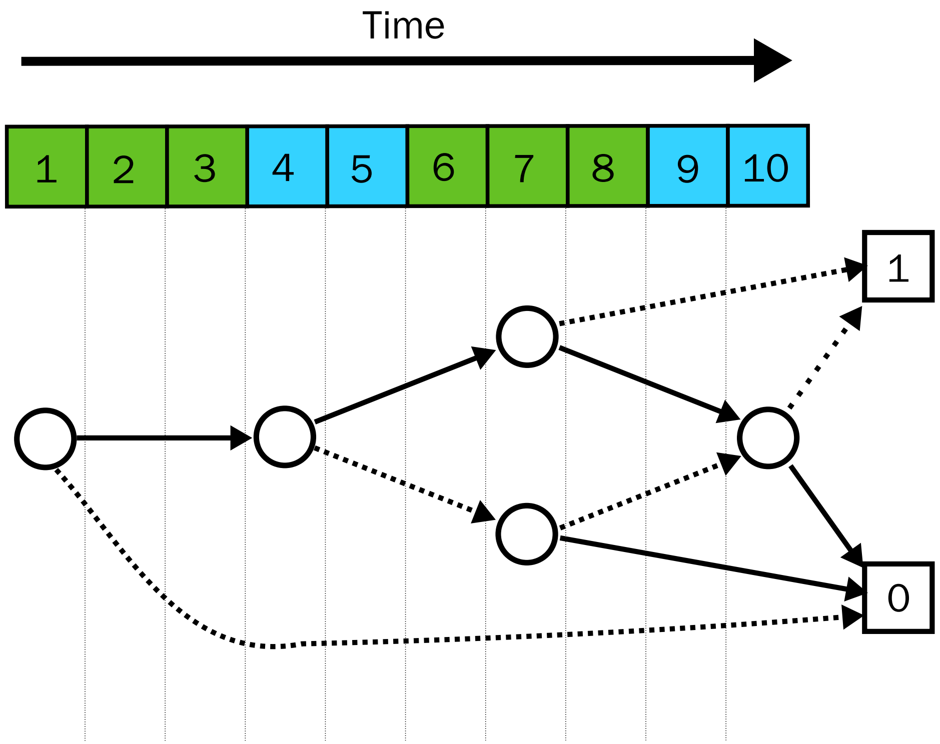
Maximum Causal Entropy and BDDs

Q: Can Maximum Entropy Causal Policy be computed on causally ordered BDDs? **A:** Yes!

1. Associativity of smax and $\mathbb{E}$.
2. $\text{smax}(\alpha, \alpha) = \alpha + \ln(2)$
3. $\text{E}(\alpha, \alpha) = \alpha$

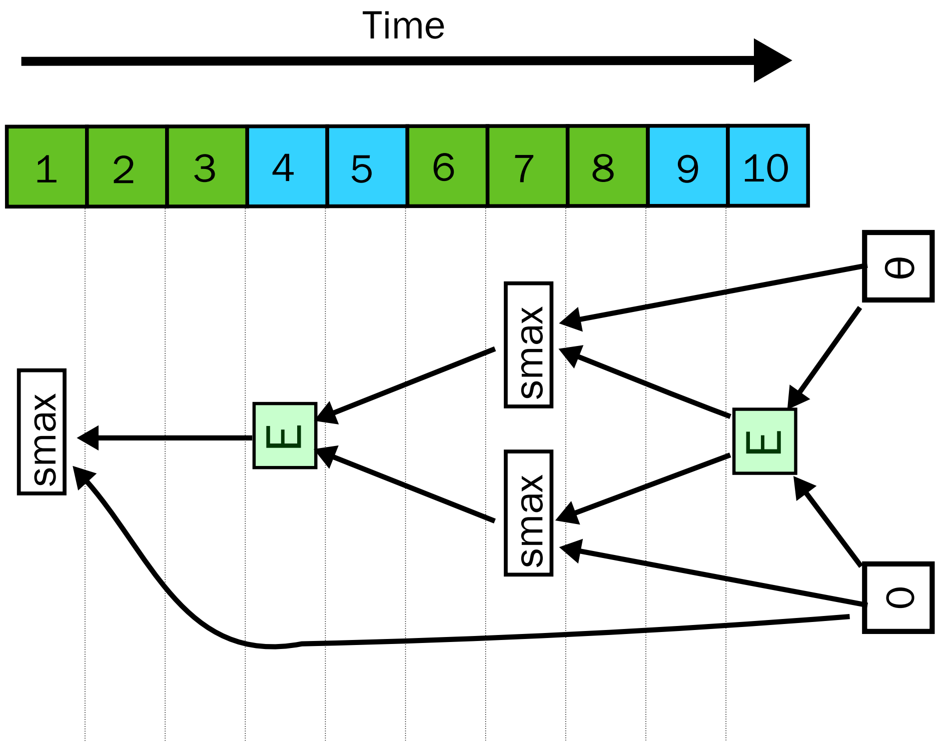
Maximum Causal Entropy and BDDs

1. Associativity of smax and $\mathbb{E}$.
2. $\mathrm{smax}(\alpha, \alpha) = \alpha + \ln(2)$
3. $\mathrm{E}(\alpha, \alpha) = \alpha$



Maximum Causal Entropy and BDDs

1. Associativity of smax and $\mathbb{E}$.
2. $\mathrm{smax}(\alpha, \alpha) = \alpha + \ln(2)$
3. $\mathrm{E}(\alpha, \alpha) = \alpha$



Size Bounds

Q: How big can these Causal BDDs be?

$$|BDD| \leq \overbrace{\underbrace{\tau}_{\text{horizon}} \cdot \big(\log(|A|) + \#\text{coins}\big)}^{\#\text{ inputs}} \cdot \big(\underbrace{|S/\varphi| \cdot |A|}_{\text{composed automaton}} \cdot 2^{\#\text{coins}} \big)$$

Size Bounds

$$|BDD| \leq \underbrace{\tau}_{\text{horizon}} \cdot \overbrace{\left(\log(|A|) + \#\text{coins}\right)}^{\#\text{ inputs}} \cdot \left(\underbrace{|S/\varphi| \cdot |A|}_{\text{composed automaton}} \cdot 2^{\#\text{coins}} \right)$$

Linear in horizon!

Note: Using function composition, can build BDD efficiently.

Max Entropy and Max Causal Entropy

a.

$$H(A_{1:\tau} \,||\, S_{1:\tau}) \approx H(A_{1:\tau} \mid S_{1:\tau})$$

Need to compute performance of unifomly random actions.

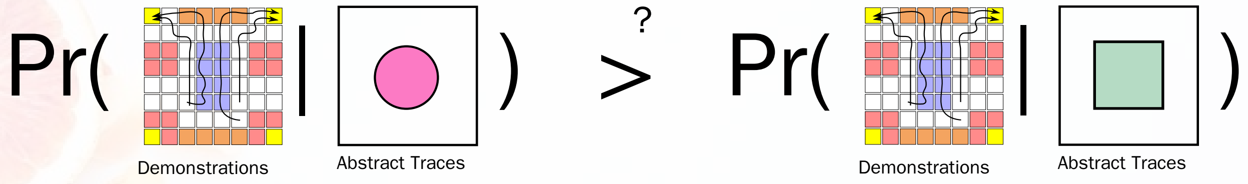
b.

$$H(A_{1:\tau} \,||\, S_{1:\tau}) \not\approx H(A_{1:\tau} \mid S_{1:\tau})$$

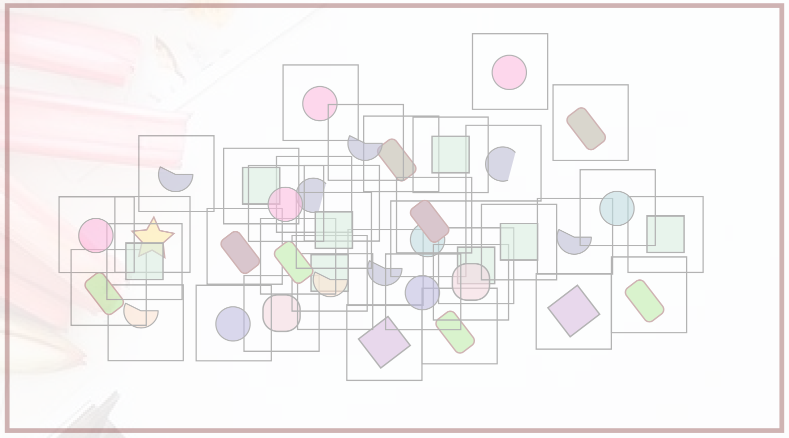
Compressed Bellman backup on binary decision diagram.

Solution Ingredients

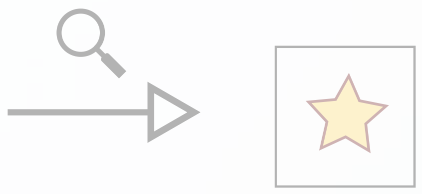
1. Compare Likelihoods.



2. Search for likely specifications.



Concept Class



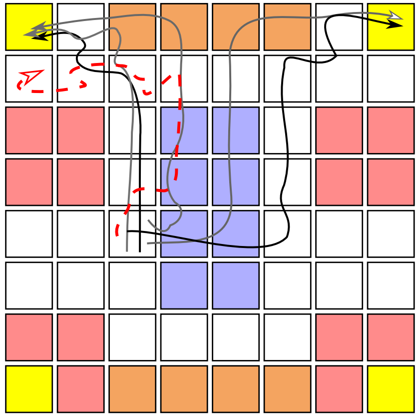
Structure of the talk

Prelude - Problem Setup

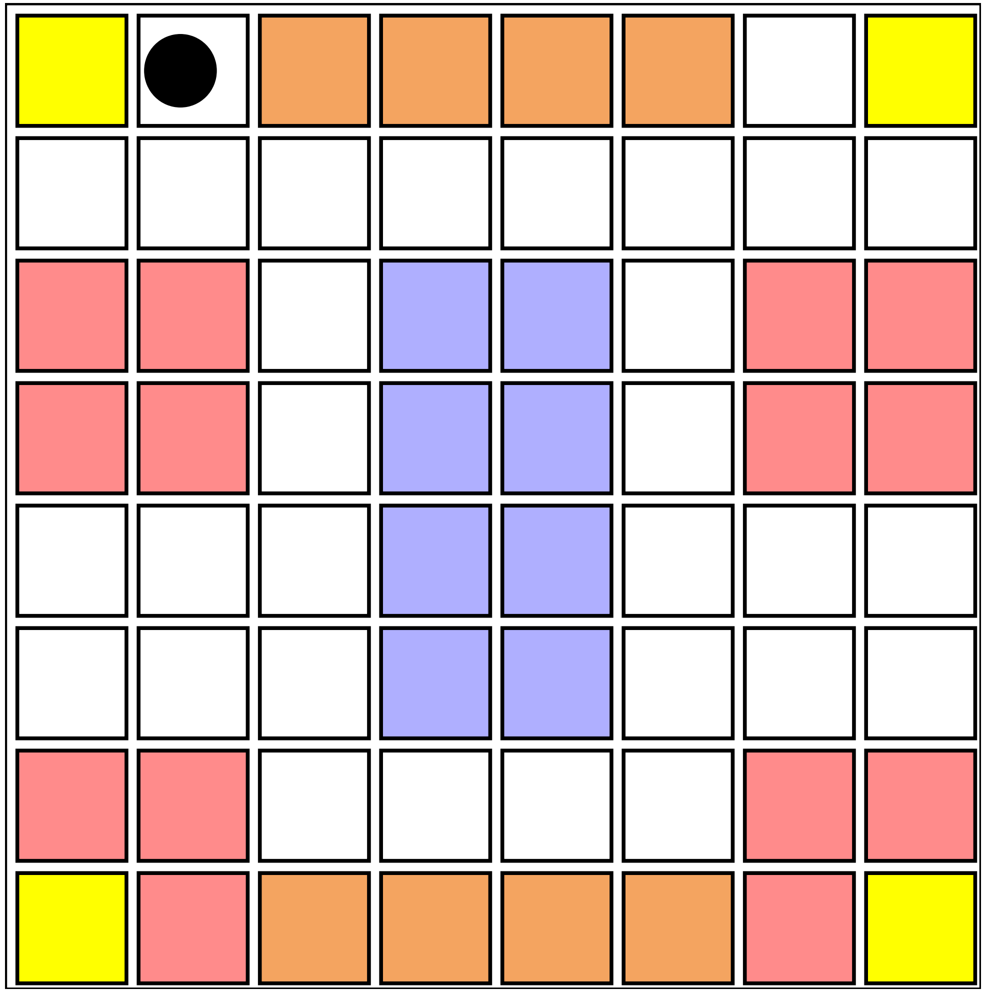
Act 1 - Naive Reduction to Maximum Causal Entropy IRL

Act 2 - Exploiting Boolean structure

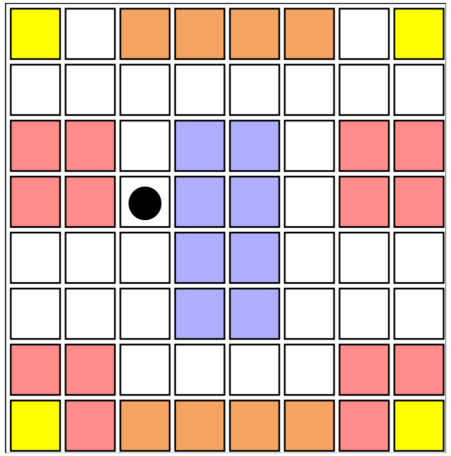
Finale - Experiment



Toy Experiment



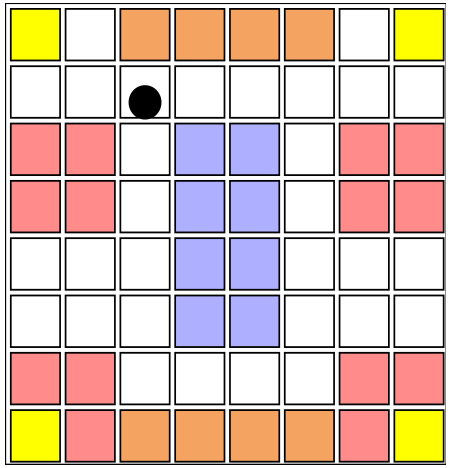
Toy Experiment



Dynamics

- Agent can attempt to move $\{\uparrow, \downarrow, \leftarrow, \rightarrow\}$.
- With probability $\frac{1}{32}$, agent will slip and move $\leftarrow$.

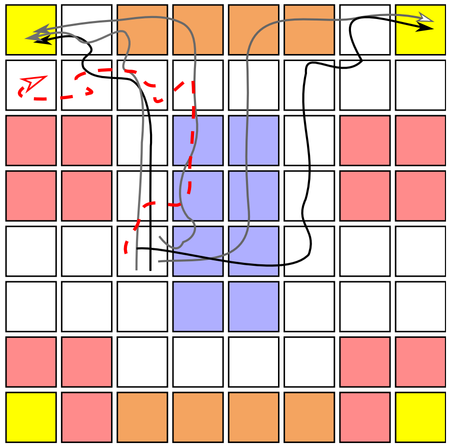
Toy Experiment



Dynamics

- A = { $\uparrow$, $\downarrow$, $\leftarrow$, $\rightarrow$ }.
- $p = \frac{1}{32}$, slip and move $\leftarrow$.

Toy Experiment



Dynamics

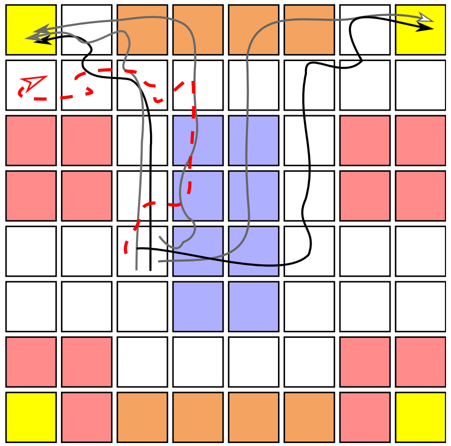
- A = { ↑ , ↓ , ←, →}.
- $p = \frac{1}{32}$, slip and move ←.

Provided 6 unlabeled demonstrations for the task:

- Go to and stay at the yellow tile (recharge).
- Avoid red tiles (lava).
- If you enter a blue, touch a brown tile **before** recharging.
- Within 10 time steps.

Note: Dashed demonstration fails to dry off due to slipping.

Toy Experiments



Dynamics

- A = { ↑ , ↓ , ←, →}.
- $p = \frac{1}{32}$, slip and move ←.

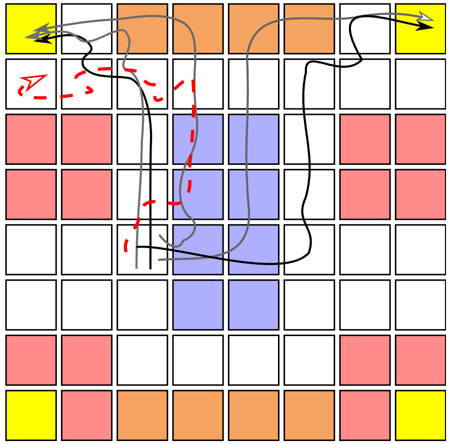
Spec	Policy Size	ROBDD	Relative Log Likelihood
	(#nodes)	build time	(Compared to True)
true	1	0.48s	0
R_1 = Avoid Lava	1797	1.5s	-22
R_2= Recharge	1628	1.2s	5
R_3= Don't recharge while wet	750	1.6s	-10
$R_4 = R_1 \wedge R_2$	523	1.9s	4
$R_5 = R_1 \wedge R_3$	1913	1.5s	-2
$R_6 = R_2 \wedge R_3$	1842	2s	15
$R_\star = R_1 \wedge R_2 \wedge R_3$	577	1.6	27
	(smaller better)	(smaller better)	(bigger better)

Toy Experiments

Spec	Policy Size	ROBDD	Relative Log Likelihood
	(#nodes)	build time	(Compared to True)
true	1	0.48s	0
R_1 = Avoid Lava	1797	1.5s	-22
R_2= Recharge	1628	1.2s	5
R_3= Don't recharge while wet	750	1.6s	-10
$R_4 = R_1 \wedge R_2$	523	1.9s	4
$R_5 = R_1 \wedge R_3$	1913	1.5s	-2
$R_6 = R_2 \wedge R_3$	1842	2s	15
$R_\star = R_1 \wedge R_2 \wedge R_3$	577	1.6	27
	(smaller better)	(smaller better)	(bigger better)

Key observation: True specification more likely than consistent specifications.

Toy Experiments



Dynamics

- A = { ↑ , ↓ , ←, →}.
- $p = \frac{1}{32}$, slip and move ←.

Find ipython binder for experiment at:
bit.ly/2WgzDcW

Code for this paper:

github.com/mvcisback/mce-spec-inference

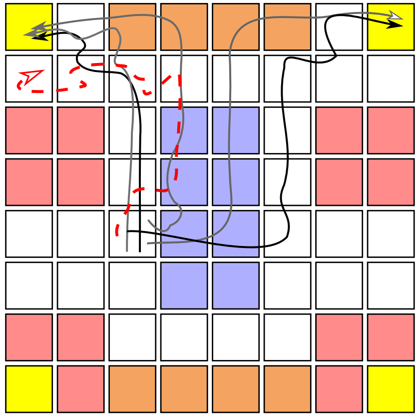
Structure of the talk

Prelude - Problem Setup

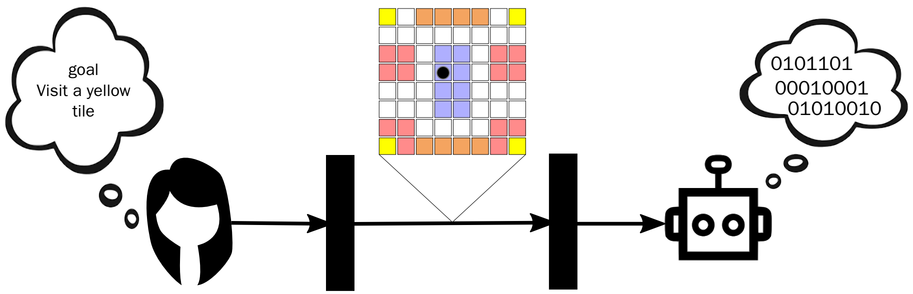
Act 1 - Naive Reduction to Maximum Causal Entropy IRL

Act 2 - Exploiting Boolean structure

Finale - Experiment



Conclusions

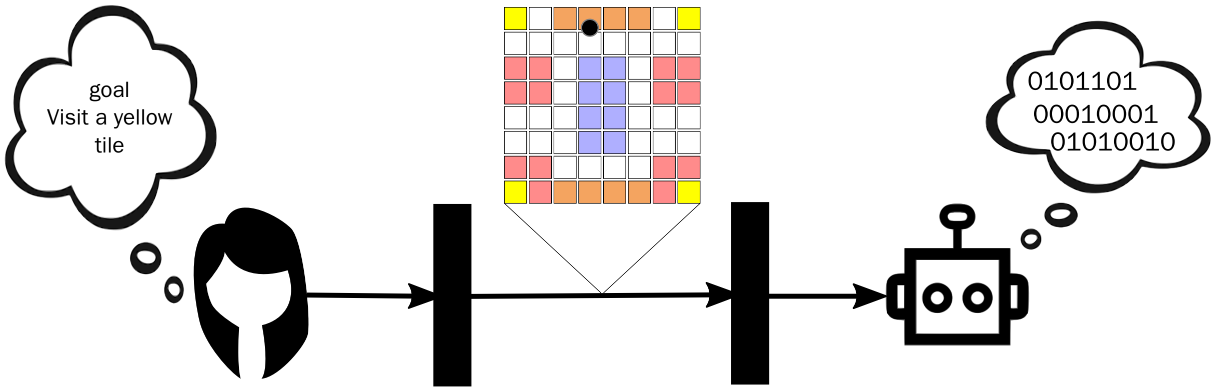


Demos are often a natural way to relay a trace property.

Can still learn given **unlabeled** demonstration errors!

Sketched 2 algorithms based on maximizing (causal) entropy.

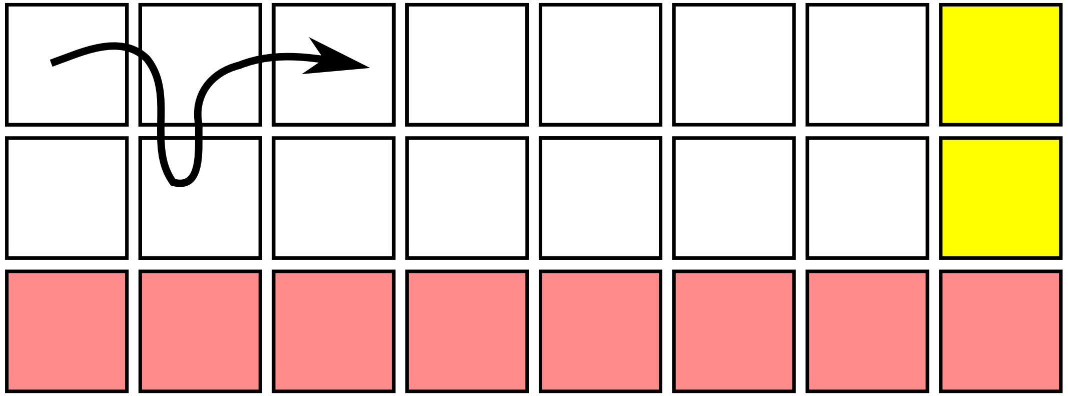
Questions?



Slides @ mjvc.me/simonsSP21

Causal Policies

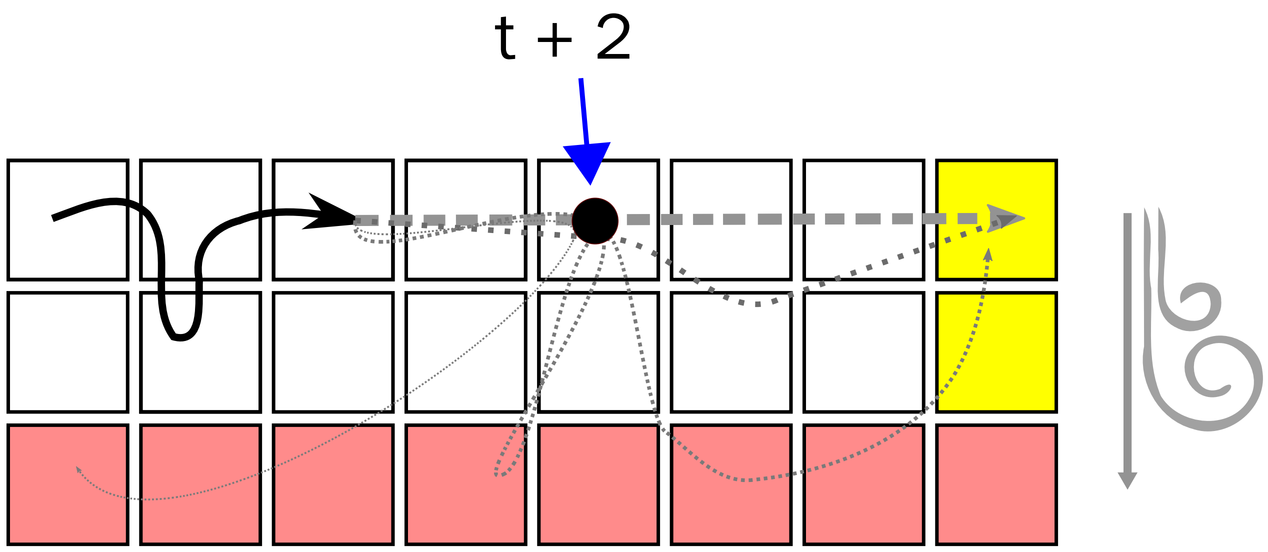
Actions shouldn't depend on information from the future.



Goal: Reach yellow. How will agent act?

Non-Causal Policies

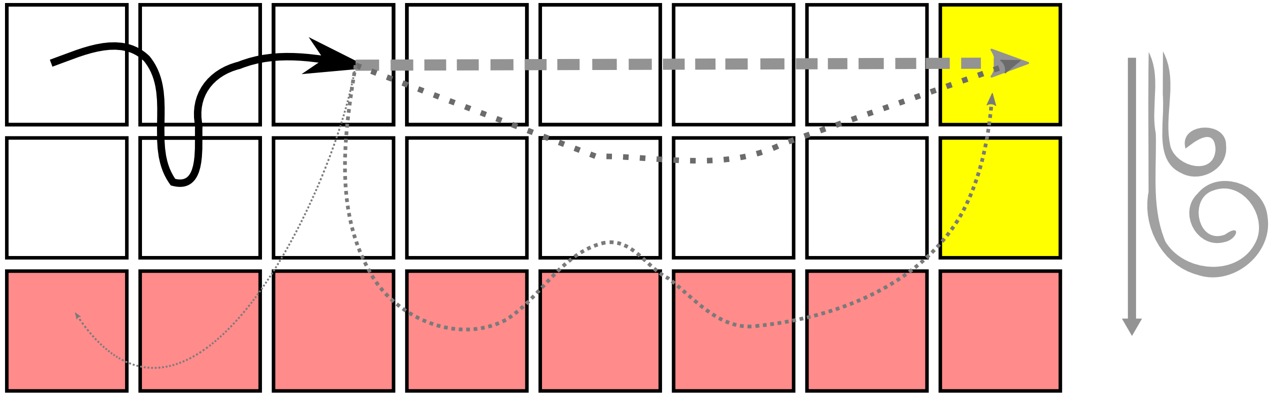
Actions shouldn't depend on information from the future.



Example of conditioning on the future.

Causal Policies

Actions shouldn't depend on information from the future.



Maybe we get pushed by wind.

Causal Conditioning

Actions shouldn't depend on information from the future.

$$\Pr(A_{1:\tau} \mid\mid S_{1:\tau}) \triangleq \prod_{t=1}^{\tau} \Pr(A_t \mid S_{1:t}, A_{1:t-1})$$

Simplify by assuming φ only depends on states.

Causal Conditioning

Actions shouldn't depend on information from the future.

$$\Pr(A_{1:\tau} \mid\mid S_{1:\tau}) = \prod_{t=1}^{\tau} \Pr(A_t \mid S_{1:t})$$

Simplify by assuming φ only depends on states.

Key problem

Given φ, was is demonstrator likely to do?

$\Pr(A_{1:\tau} \mid\mid S_{1:\tau})$ = ?

Maximum Causal Entropy

$$\Pr(A_{1:T} \,||\, S_{1:T}) = \ ?$$

Key Idea: Don't commit more than the observations require.

Formally: Maximize expected causal entropy.

$$H(A_{1:T} \,||\, S_{1:T}) \triangleq \mathbb{E}\left[\log\left(\frac{1}{\Pr(A_{1:T} \,||\, S_{1:T})}\right)\right]$$

subject to $\mathbb{E}[r(S_{1:T})] = r^*$.

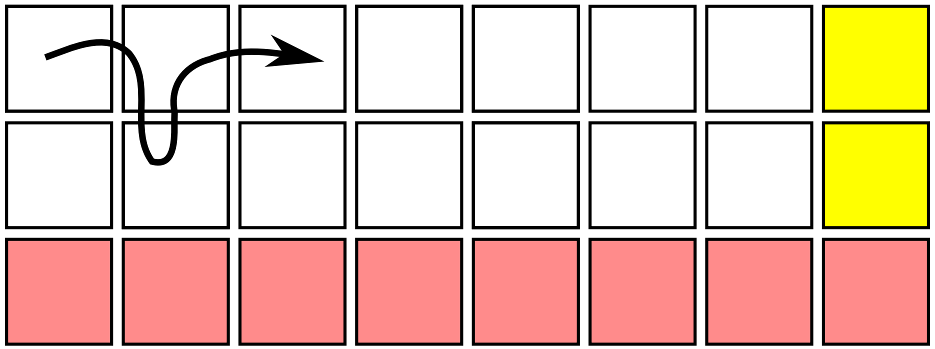
High Entropy Policies are Robust

Maximize

$$H(A_{1:\tau} \mid\mid S_{1:\tau}) \triangleq \mathbb{E}\left[\log\left(\frac{1}{\Pr(A_{1:\tau} \mid\mid S_{1:\tau})}\right)\right]$$

while matching satisification probabilities.

Goal: Reach yellow. How will agent act?



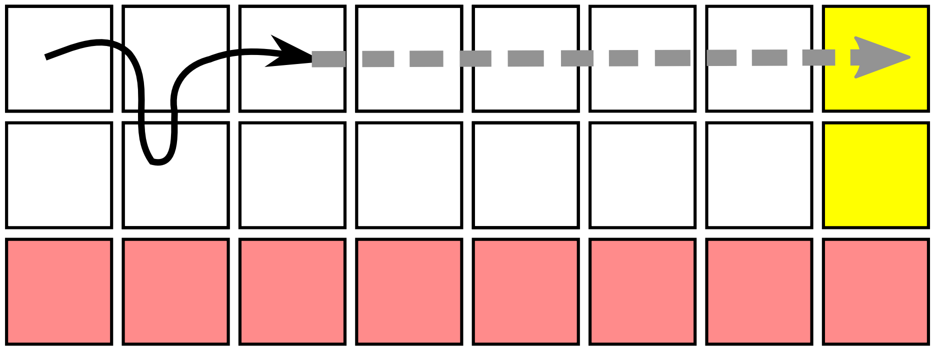
High Entropy Policies are Robust

Maximize

$$H(A_{1:\tau} \mid\mid S_{1:\tau}) \triangleq \mathbb{E}\left[\log\left(\frac{1}{\Pr(A_{1:\tau} \mid\mid S_{1:\tau})}\right)\right]$$

while matching satisification probabilities.

Minimum Entropy Forecaster



Put all of the probability mass one 1 path.

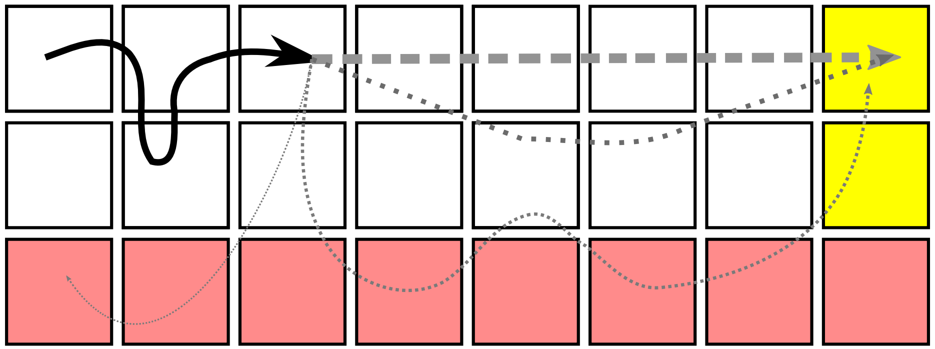
High Entropy Policies are Robust

Maximize

$$H(A_{1:\tau} \mid\mid S_{1:\tau}) \triangleq \mathbb{E}\left[\log\left(\frac{1}{\Pr(A_{1:\tau} \mid\mid S_{1:\tau})}\right)\right]$$

while matching satisification probabilities.

High Entropy Forecaster



Distribute prediction over high value paths.