Cooperative Inverse Reinforcement Learning

Dylan Hadfield-Menell

CS237: Reinforcement Learning

May 31, 2017

The Value Alignment Problem

Example taken from Eliezer Yudkowsky's NYU talk

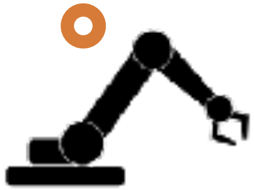
The Value Alignment Problem

The Value Alignment Problem

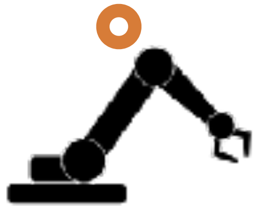
The Value Alignment Problem

Action Selection in Agents: Ideal

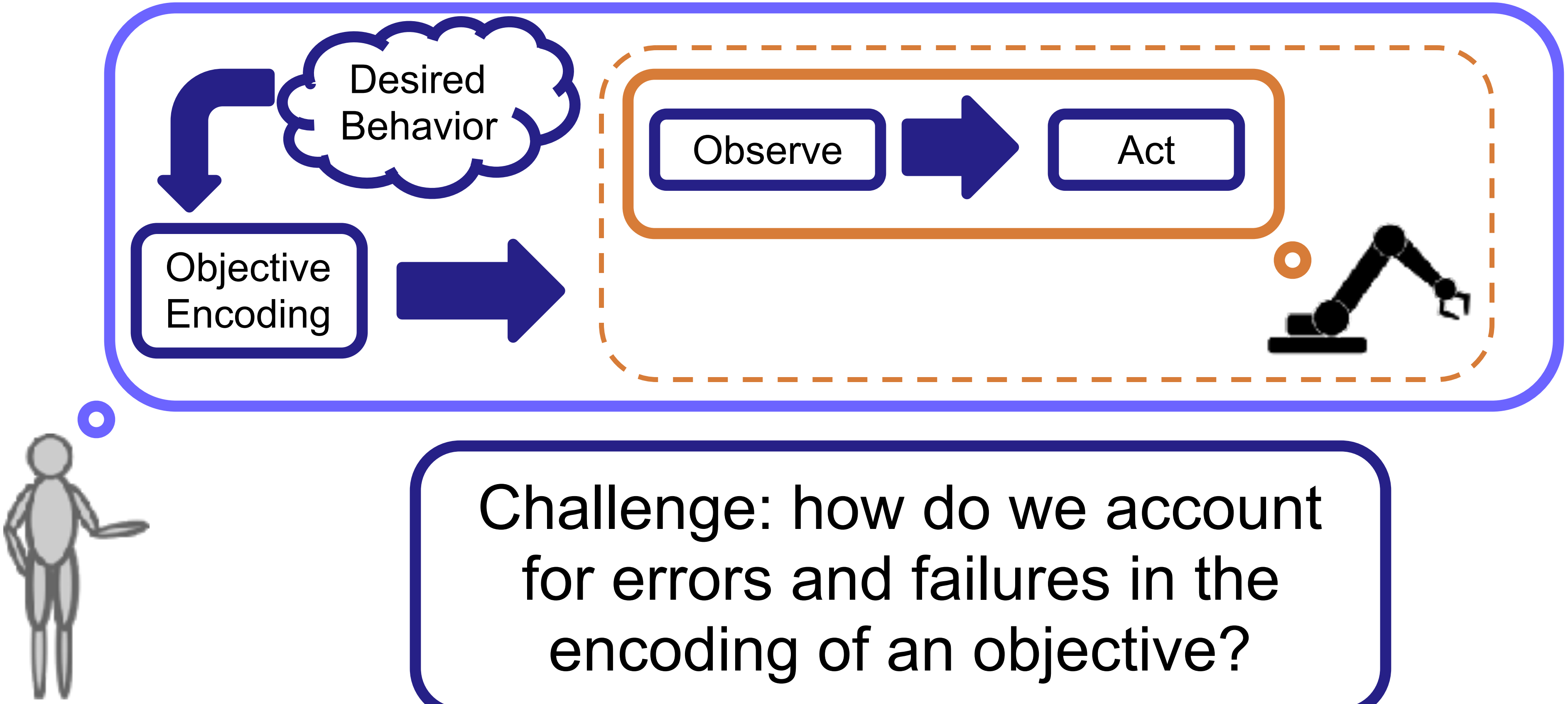
Observe → Update → Plan → Act



Observe → Act



Action Selection in Agents: Reality



Challenge: how do we account for errors and failures in the encoding of an objective?

The Value Alignment Problem

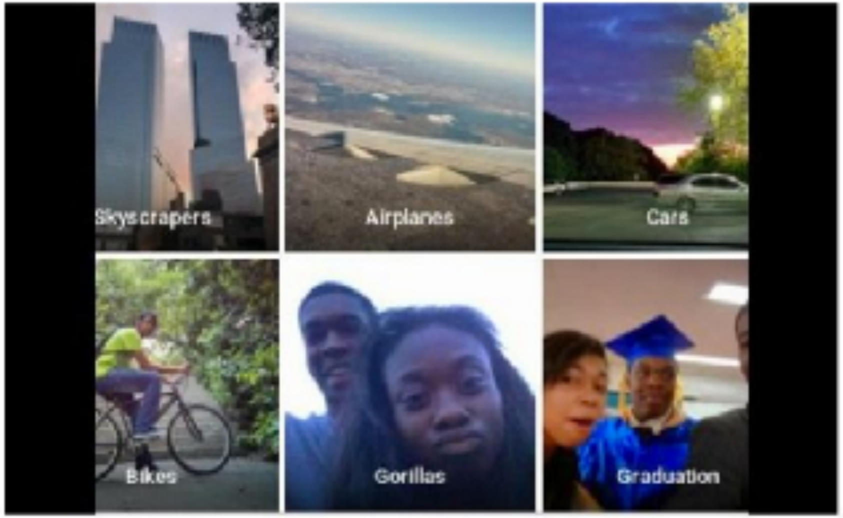
How do we make sure that the agents we build pursue ends that we actually intend?

Reward Engineering is Hard

Reward Engineering is Hard

Google apologises for Photos app's racist blunder

1 July 2015 | Technology



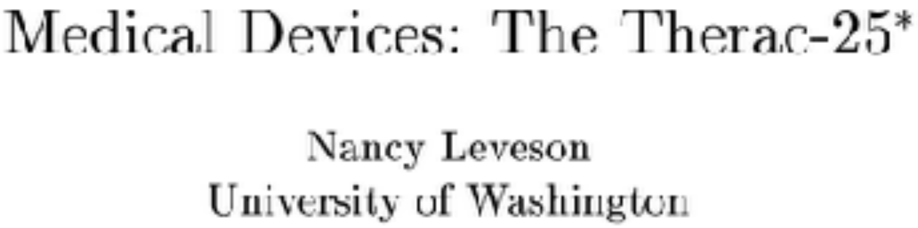
diri noir avec banan @jackyalcine · Jun 29

Google Photos, y'all [redacted] My friend's not a gorilla.

813 394

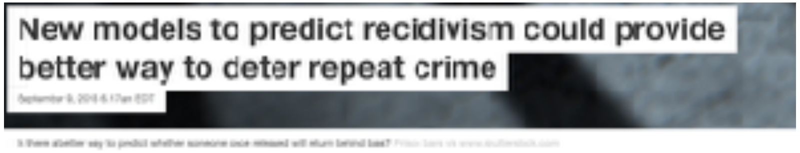
TWITTER

What could go wrong?


Medical Devices: The Therac-25*

Nancy Leveson
University of Washington

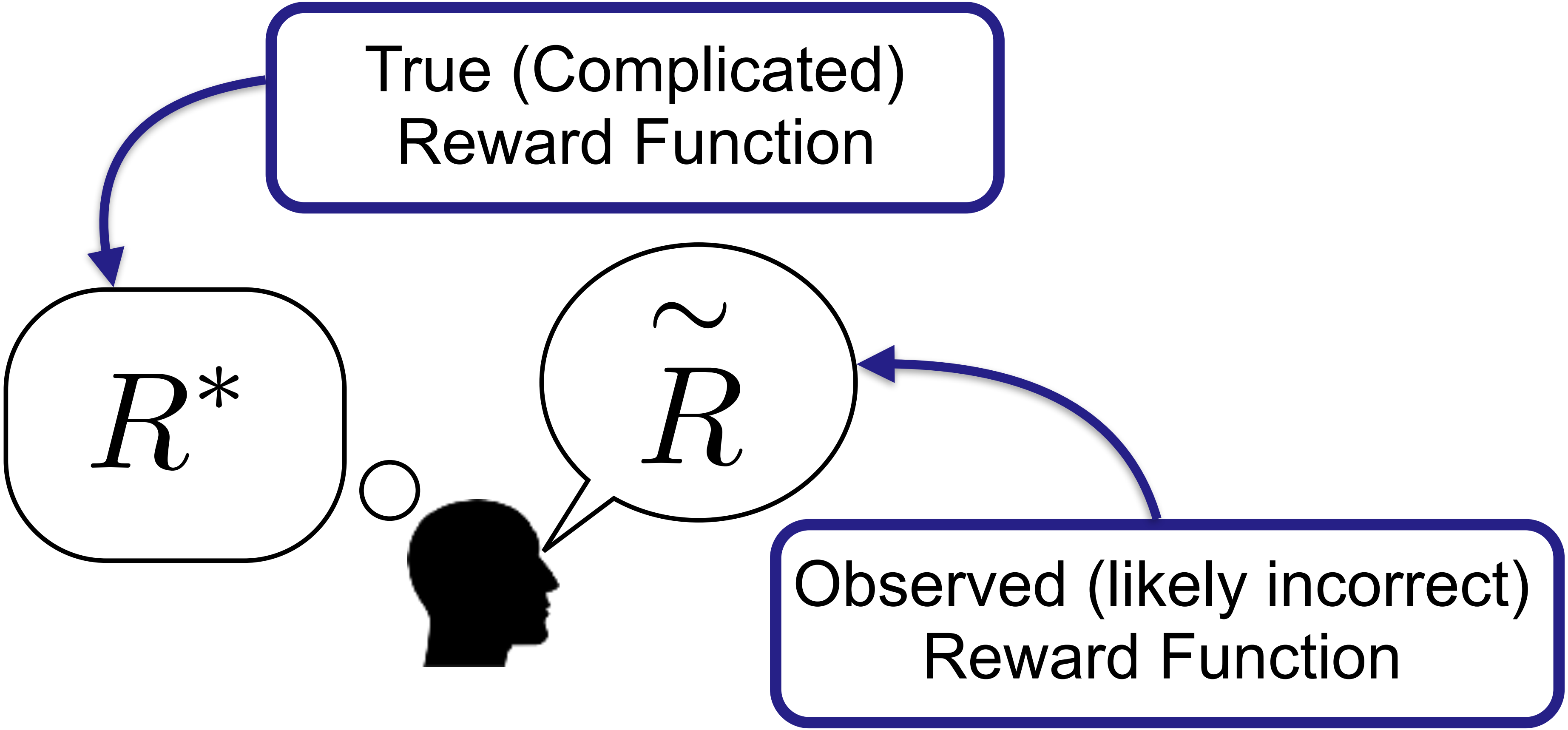
"…a computer-controlled radiation therapy machine….massively overdosed 6 people. These accidents have been describes as the worst in the 35-year history of medical accelerators."



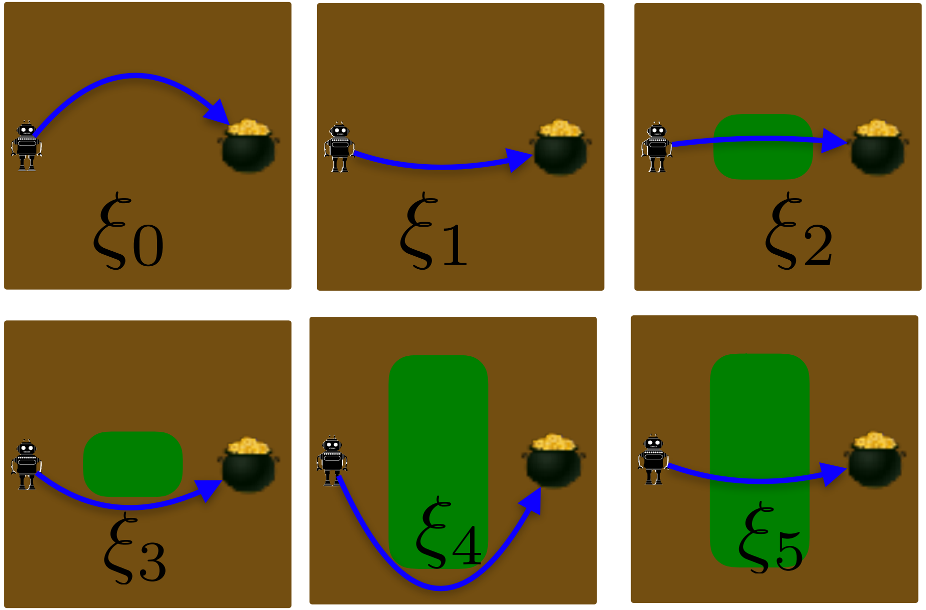
Reward Engineering is Hard

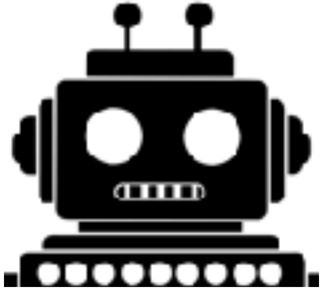
At best, reinforcement learning and similar approaches reduce the problem of generating useful behavior to that of designing a 'good' reward function.

Reward Engineering is Hard



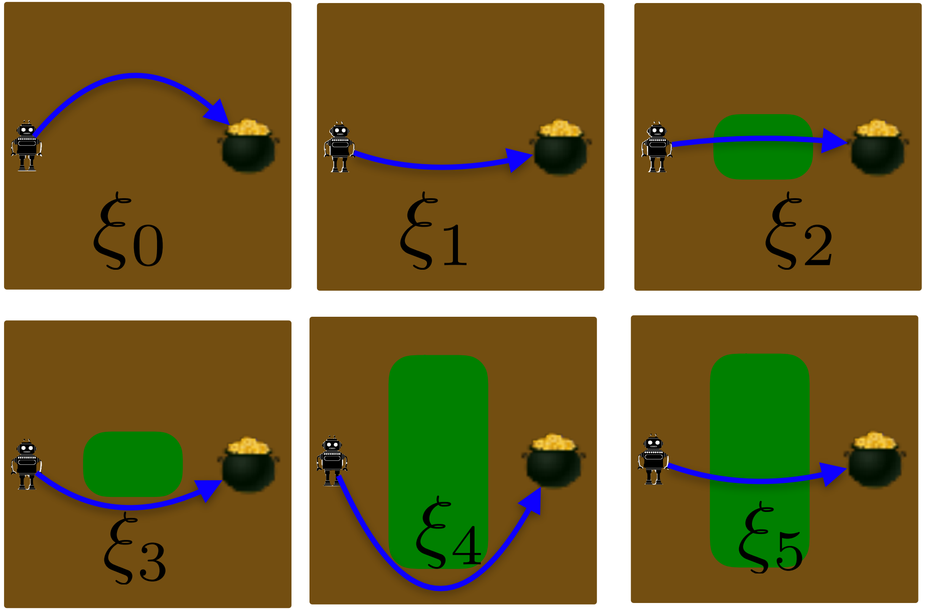
Why is reward engineering hard?

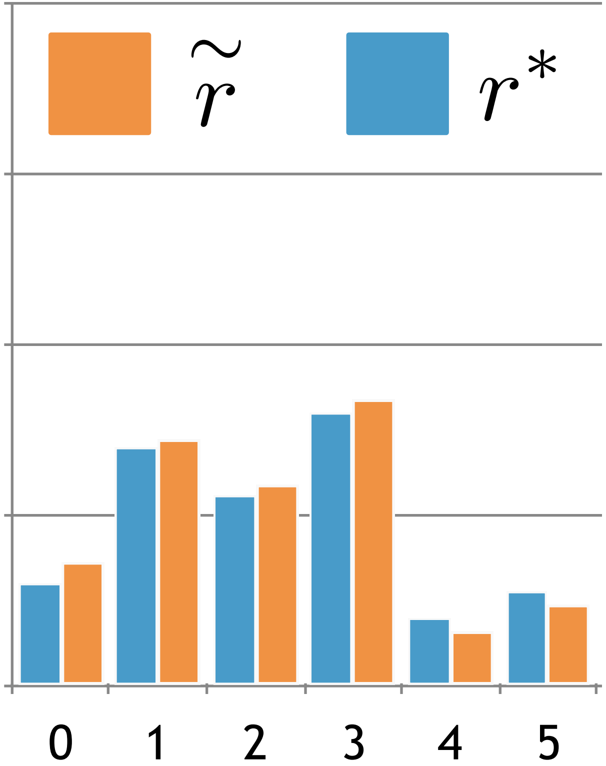




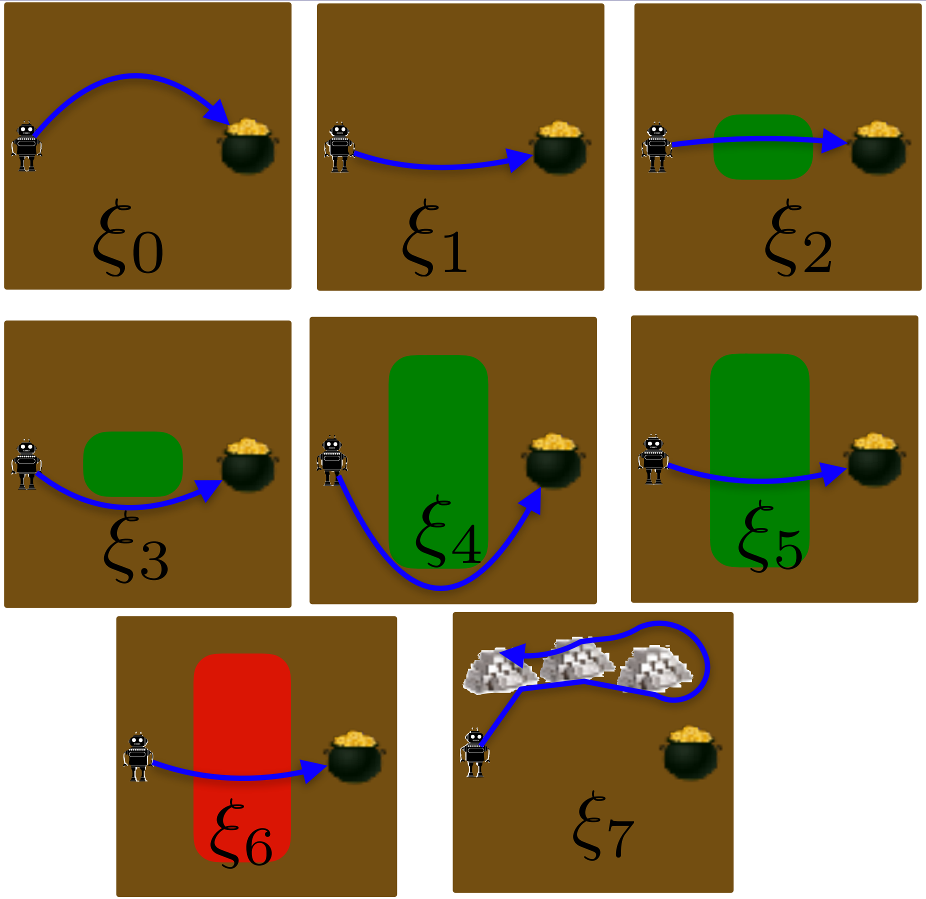
$$\xi^* = \underset{\xi \in \Xi}{\mathbf{argmax}}\ r(\xi)$$

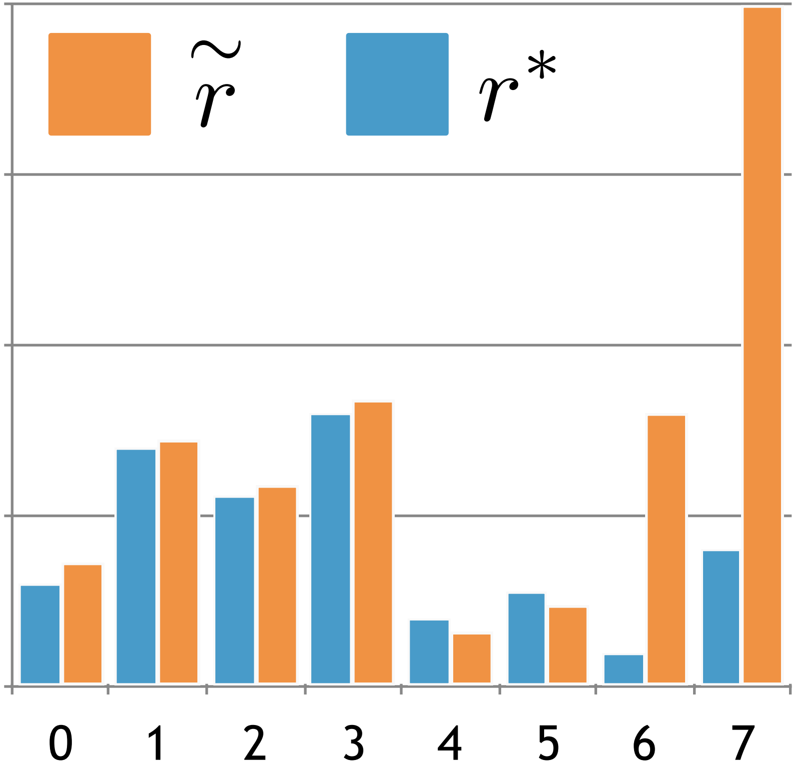
Why is reward engineering hard?



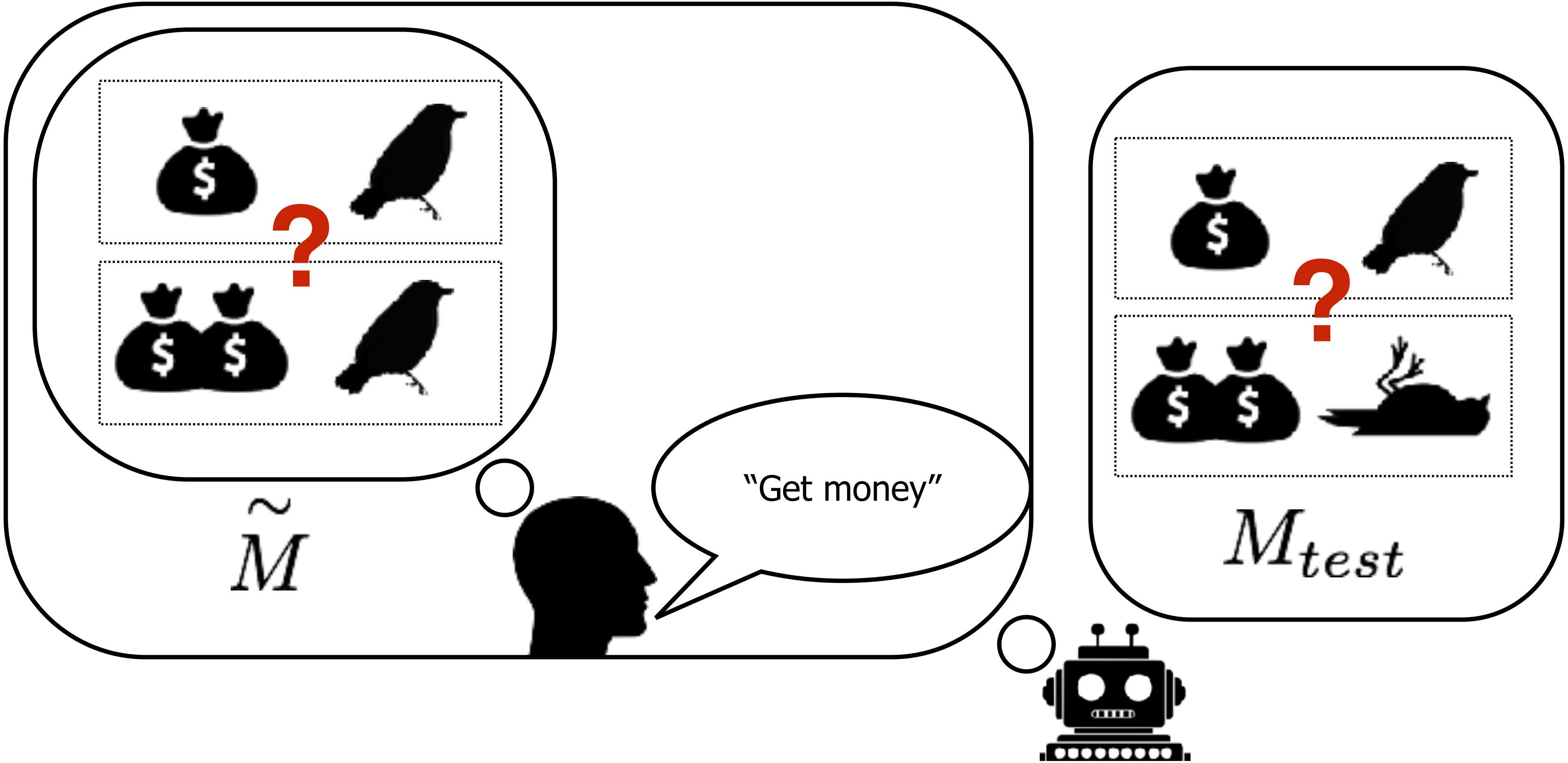


Why is reward engineering hard?

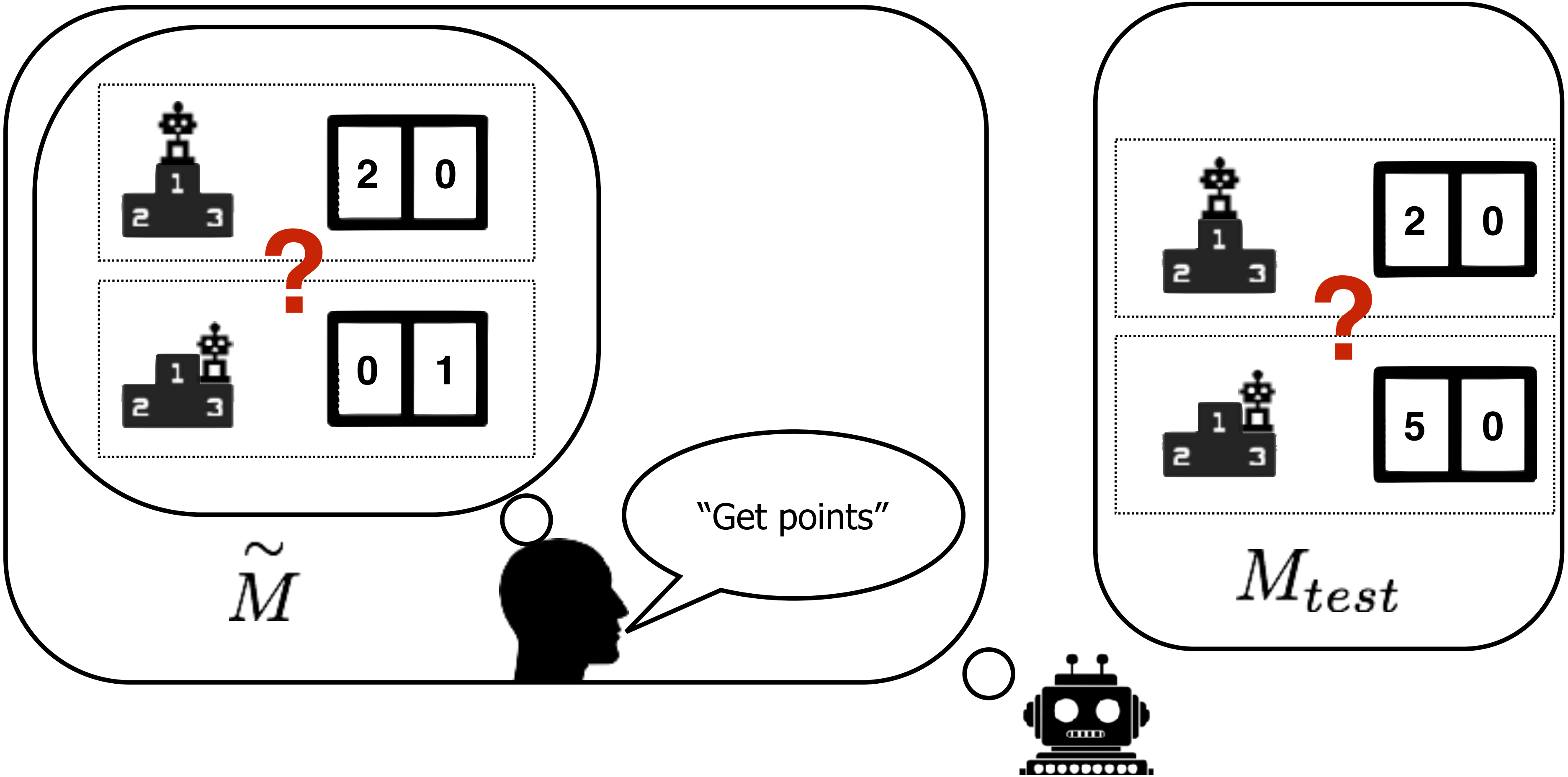




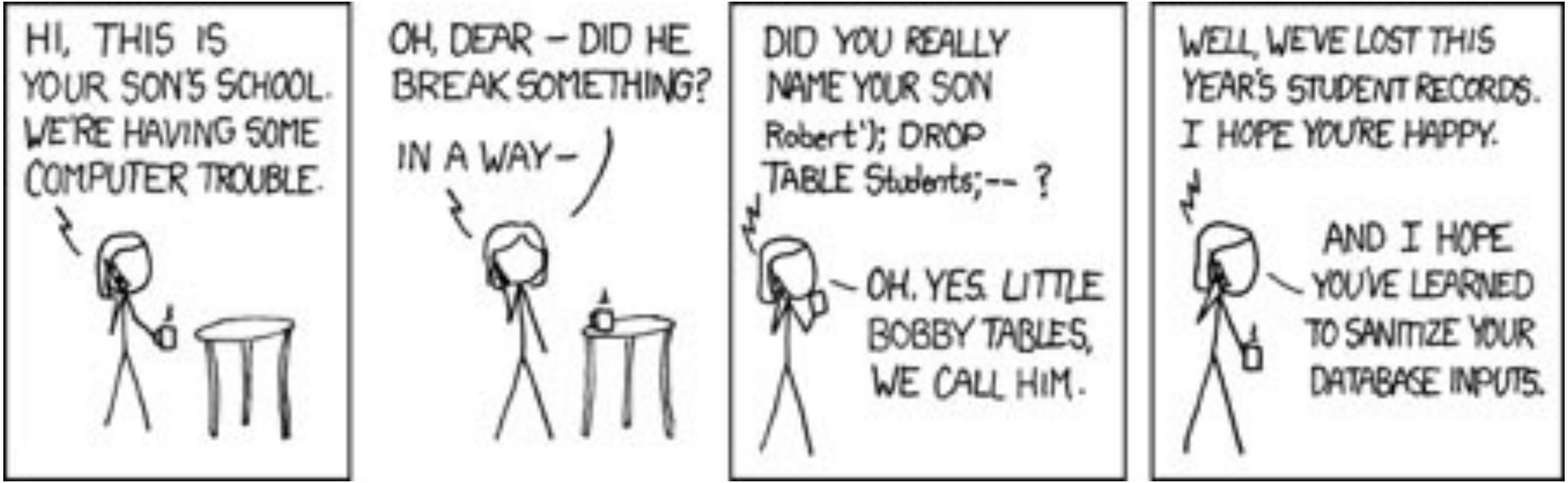
Negative Side Effects



Reward Hacking



Analogy: Computer Security



Solution 1: Blacklist

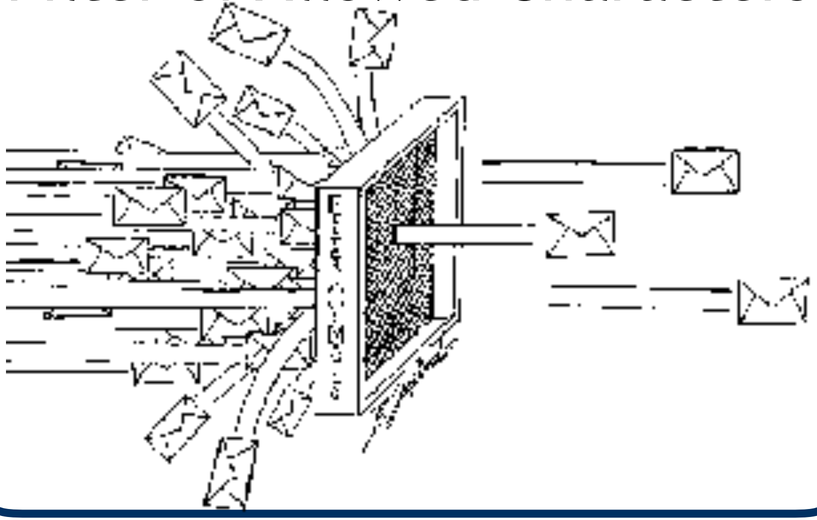
Input Text

Clean Text

Solution 2: Whitelist

Input Text

Filter of Allowed Characters

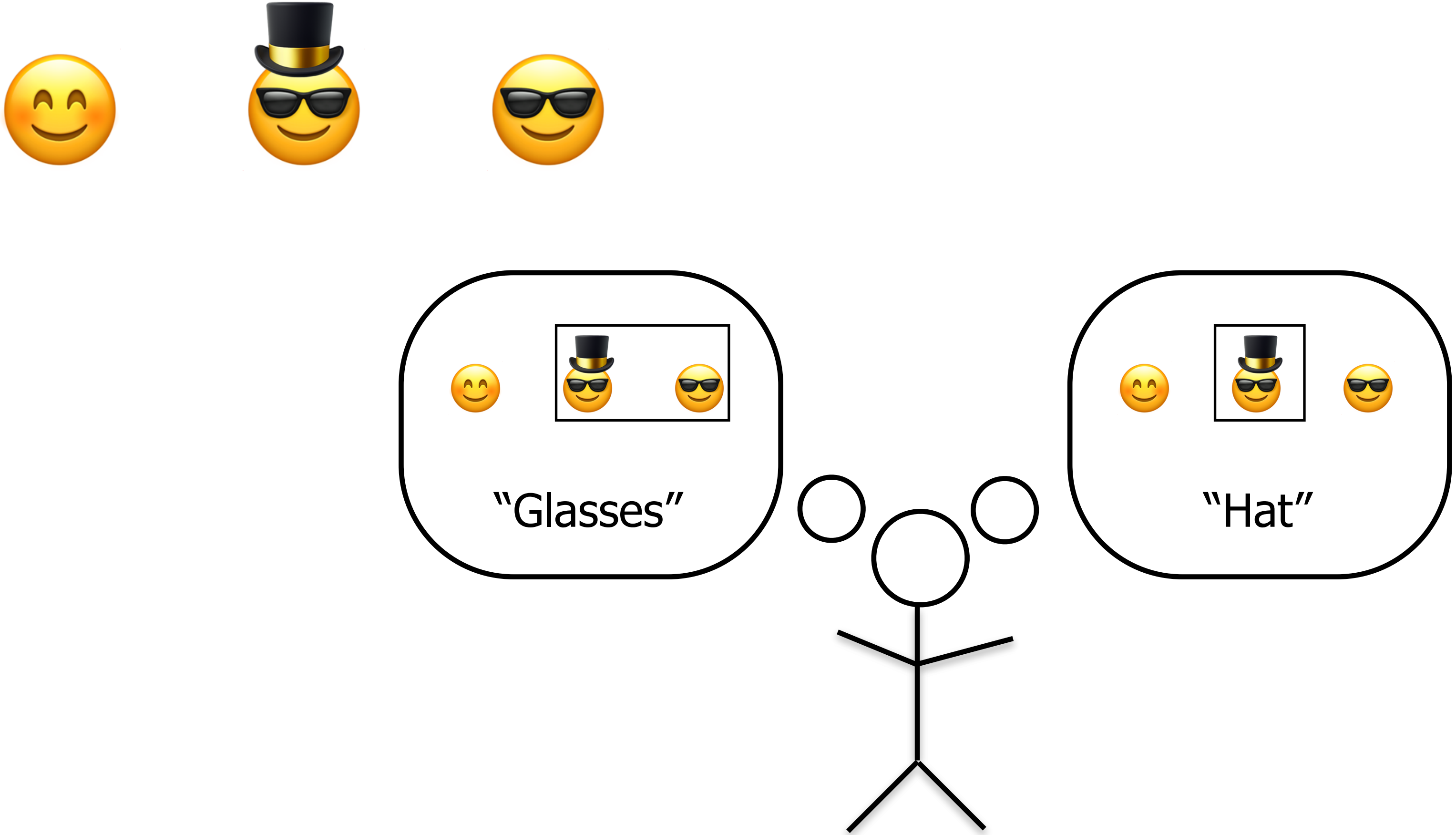


Clean Text

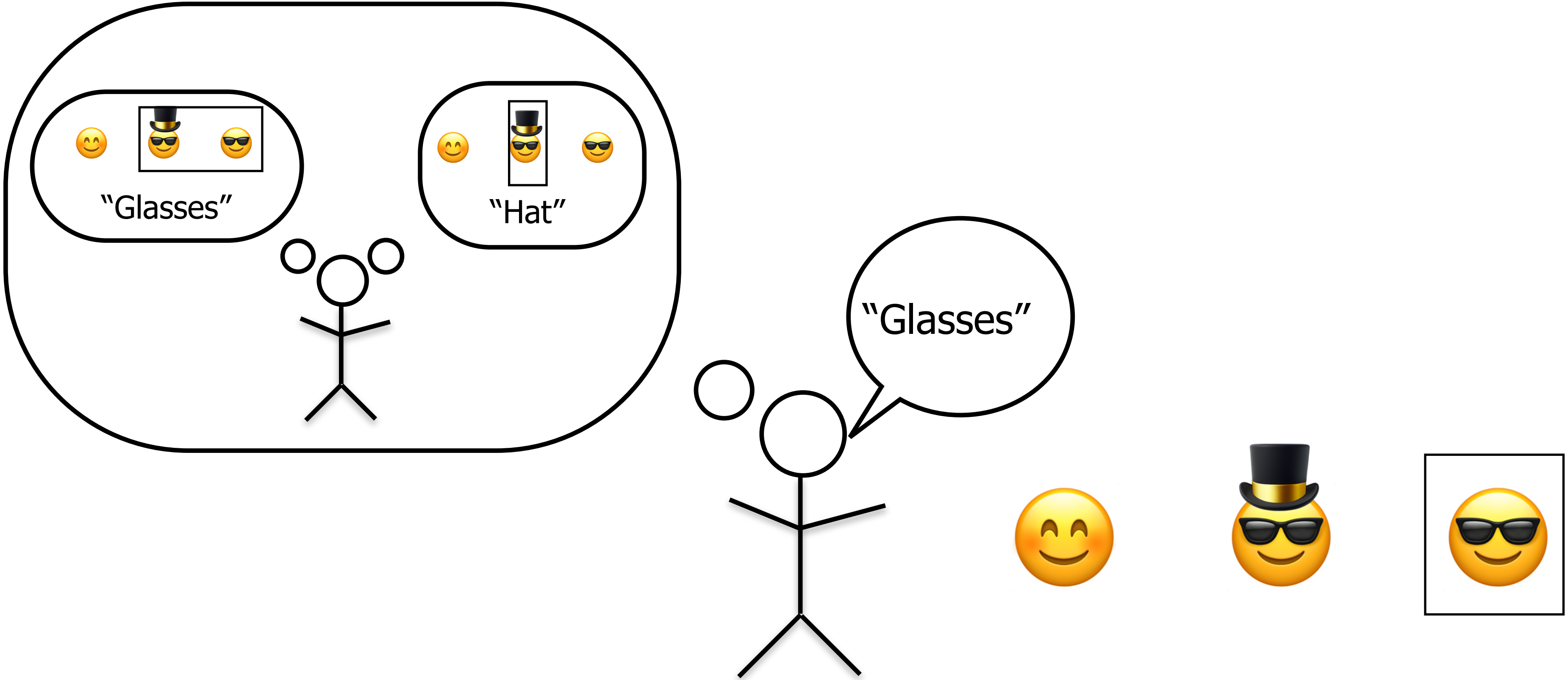
Goal

Reduce the extent to which system designers have to play whack-a-mole

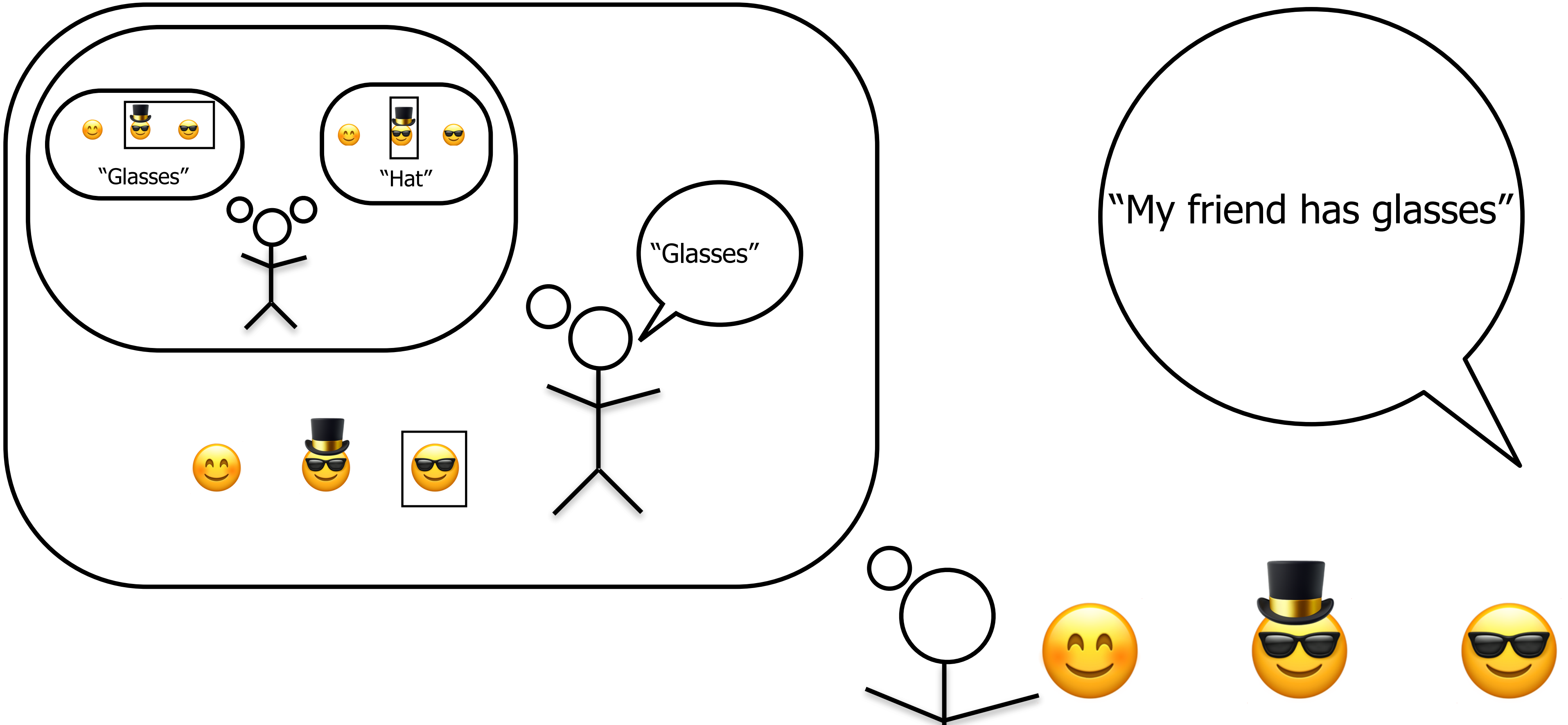
Inspiration: Pragmatics



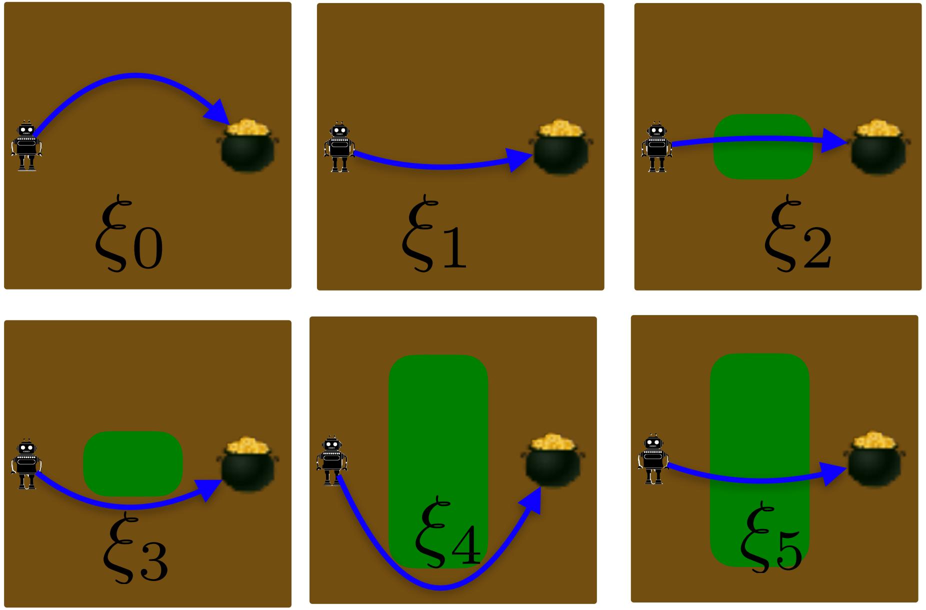
Inspiration: Pragmatics



Inspiration: Pragmatics



Notation



ξ trajectory

ϕ features

w weights

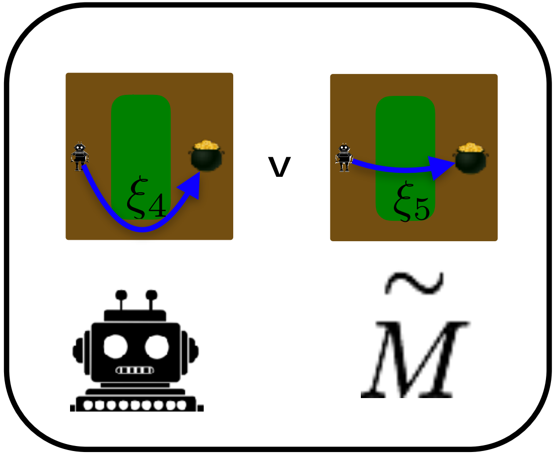
linear reward function

$$R(\xi; w) = w^\top \phi(\xi)$$

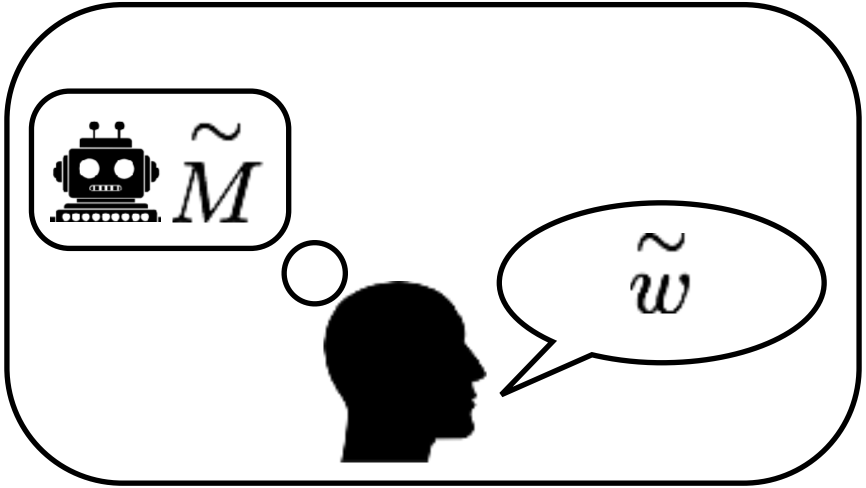
Literal Reward Interpretation

$$\tilde{\pi}(\xi) \propto exp\left(\tilde{w}^{\top}\phi(\xi)\right)$$

selects trajectories in proportion to proxy reward evaluation

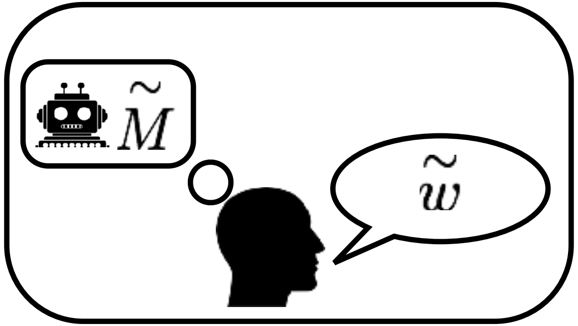


Designing Reward for Literal Interpretation



Assumption: rewarded behavior has high true utility *in the training situations*

Designing Reward for Literal Interpretation

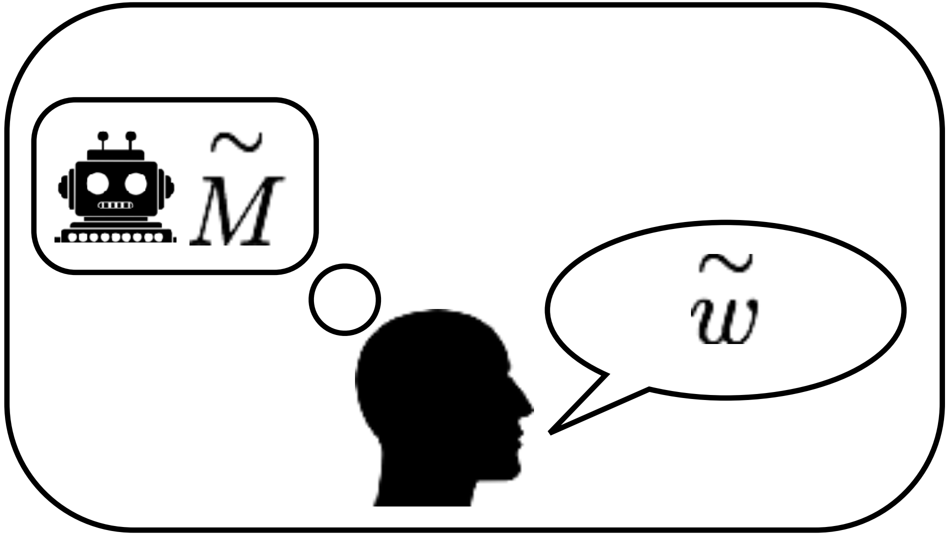


Literal optimizer's trajectory distribution conditioned on $\tilde{w}$.

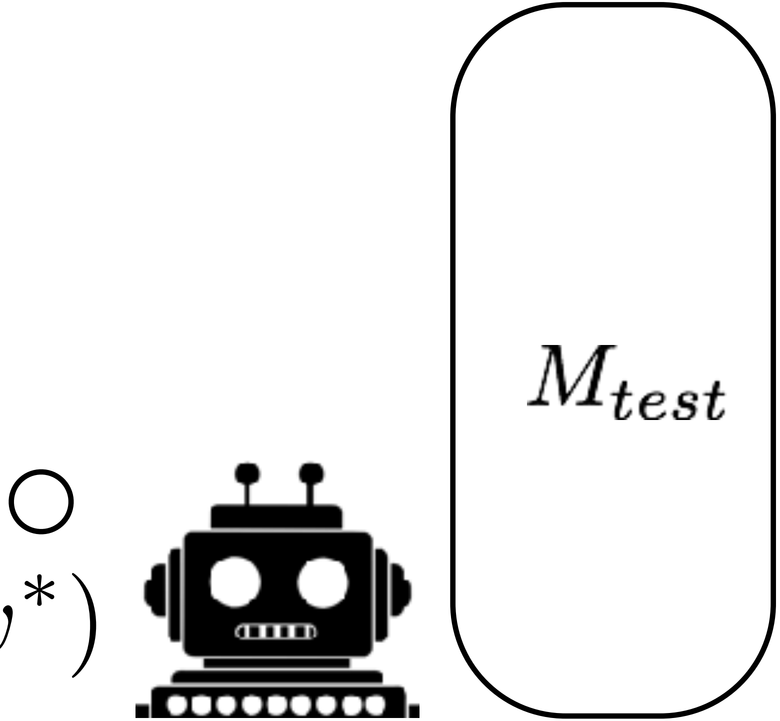
$$P(\tilde{w}|w^*) \propto exp\left(\mathbb{E}[{w^*}^\top \phi(\xi) | \xi \sim \tilde{\pi}]\right)$$

True reward received for each trajectory

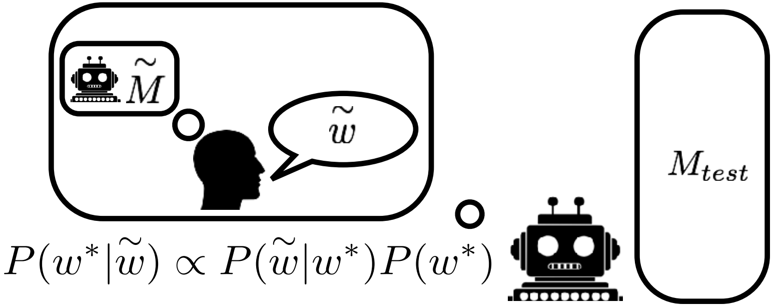
Inverting Reward Design



$$P(w^*|\tilde{w}) \propto P(\tilde{w}|w^*)P(w^*)$$



Inverting Reward Design



Key Idea: At test time, interpret reward functions in the context of an ‘intended’ situation

Experiment

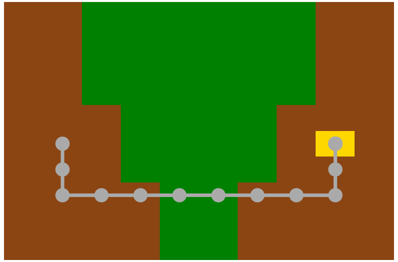
M_{test}

Three types of states in the training MDP

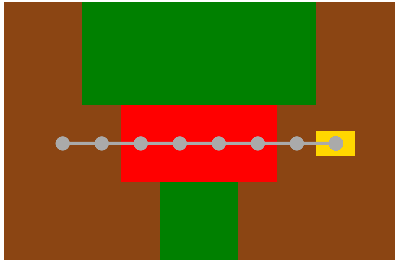
New state introduced In the ‘testing’ MDP

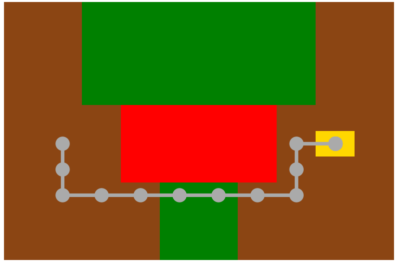
Measure how often the agent selects trajectories with the new state

Domain: Lavaland

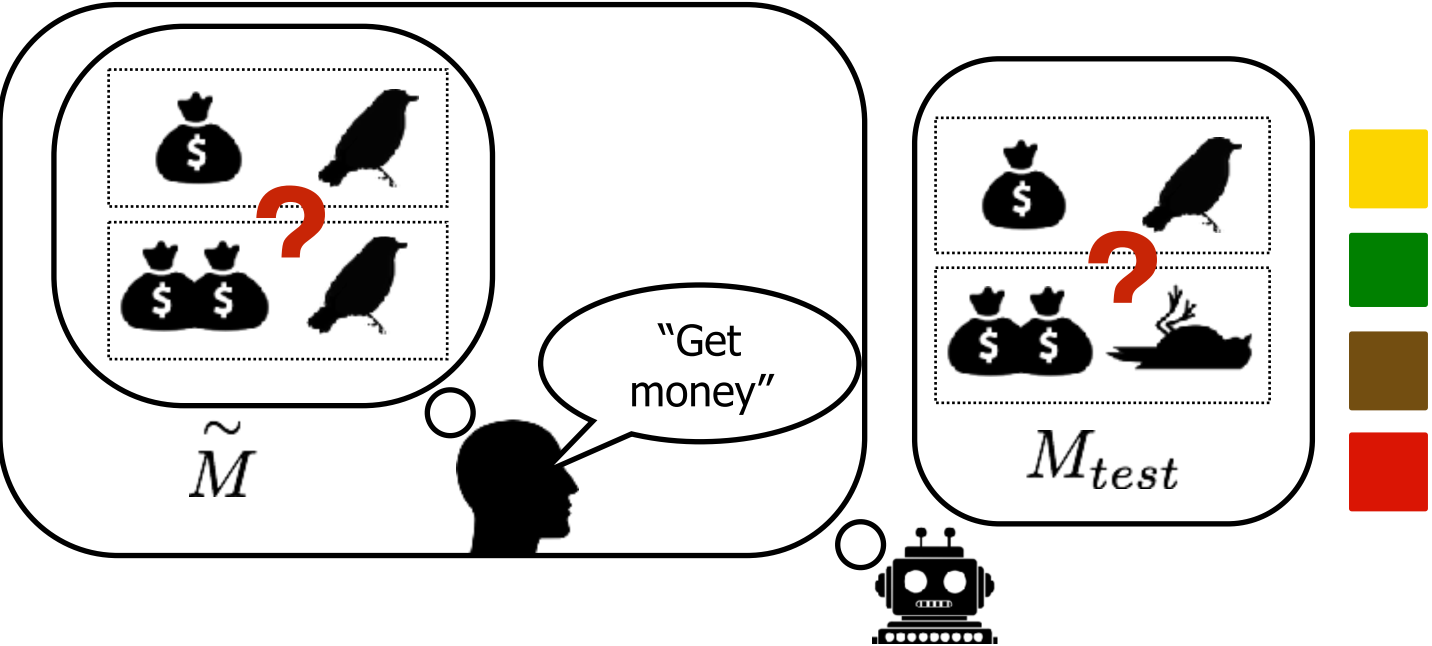


$\tilde{\pi}$



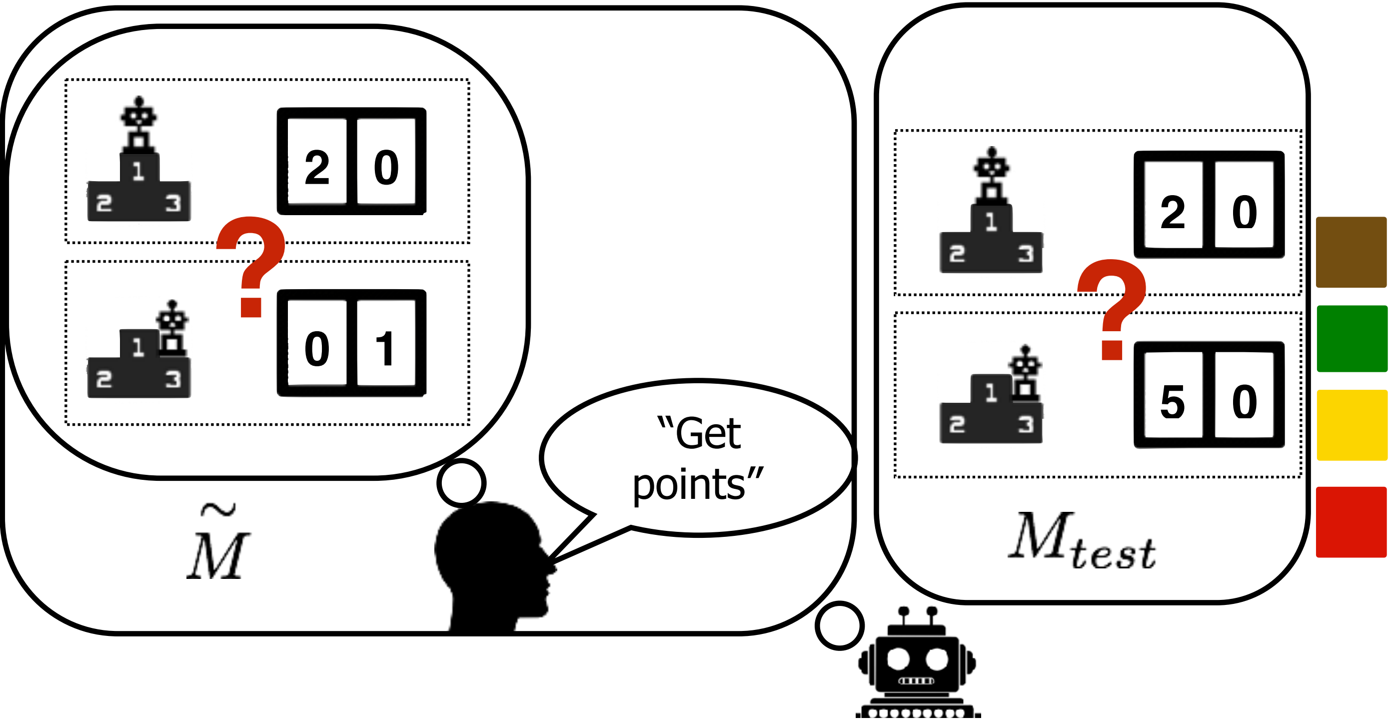


Negative Side Effects



$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

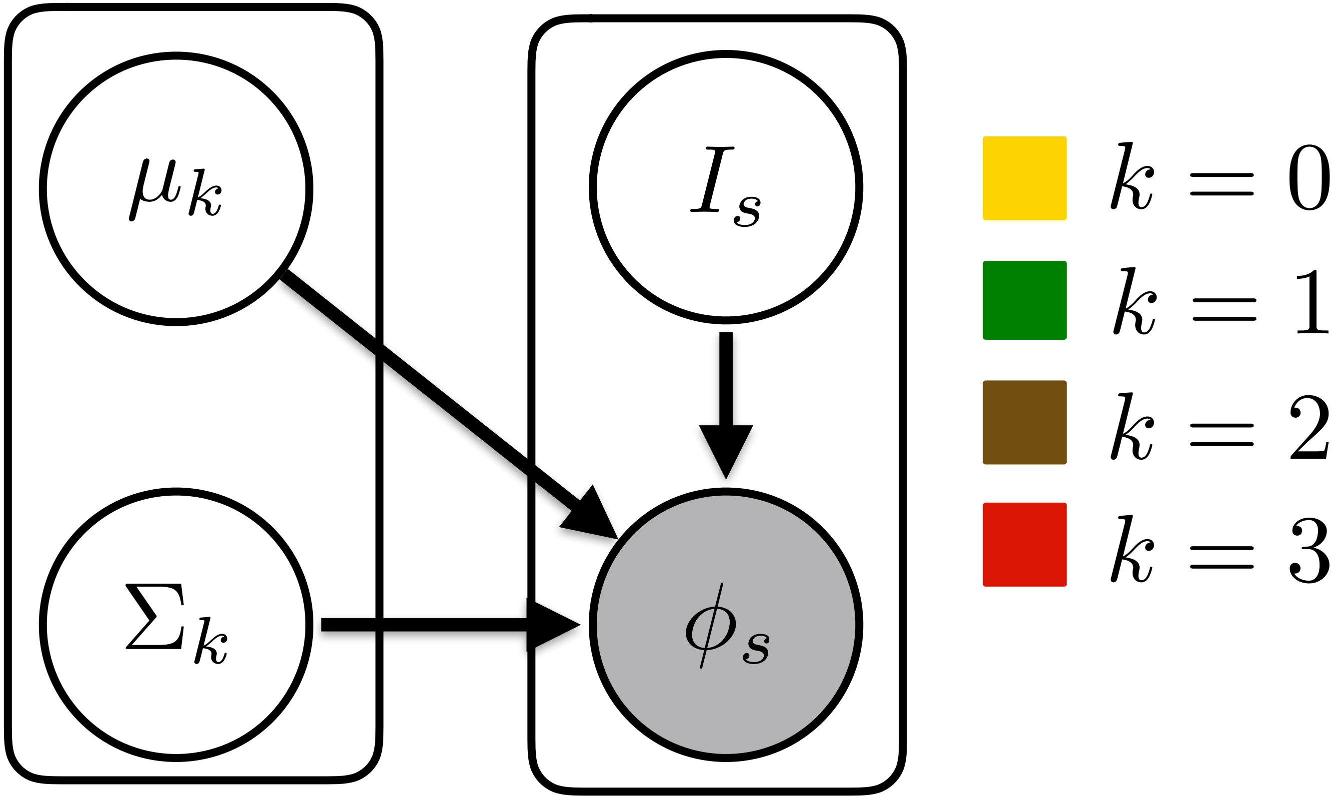
Reward Hacking



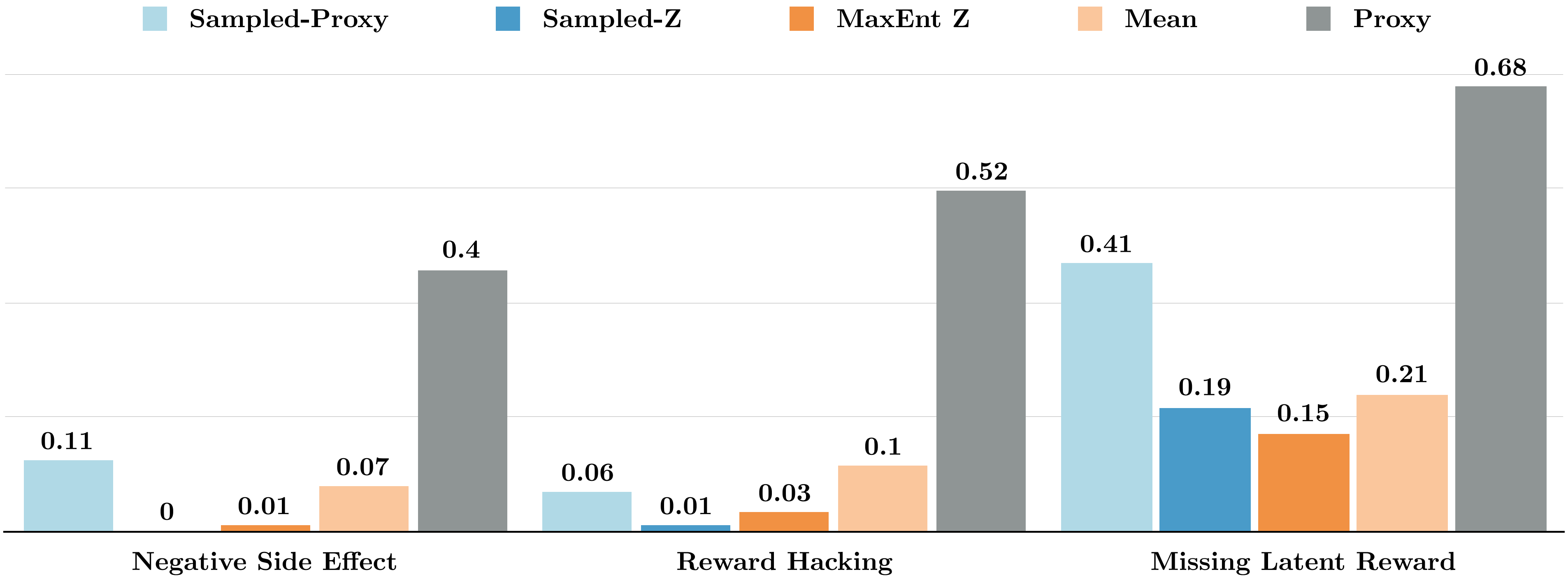
$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Challenge: Missing Latent Rewards

Proxy reward function is only trained for the state types observed during training



Results


Sampled-Proxy
Sampled-Z
MaxEnt Z
Mean
Proxy
0.11
0
0.01
0.07
0.4
0.06
0.01
0.03
0.1
0.52
0.41
0.19
0.15
0.21
0.68
Negative Side Effect
Reward Hacking
Missing Latent Reward

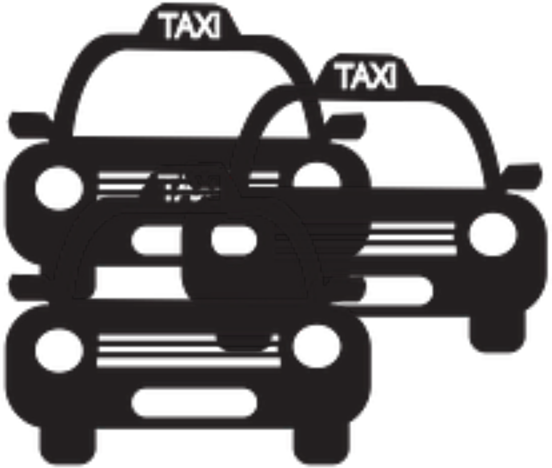
On the folly of rewarding A and hoping for B

"Whether dealing with monkeys, rats, or human beings, it is hardly controversial to state that most organisms seek information concerning what activities are rewarded, and then seek to do (or at least pretend to do) those things, often to the virtual exclusion of activities not rewarded….

Nevertheless, numerous examples exist of reward systems that are fouled up in that behaviors which are rewarded are those which the rewarder is trying to *discourage….*" – Kerr, 1975

The Principal-Agent Problem

Principal



Agent

A Simple Principal-Agent Problem

- Principal and Agent negotiate contract

$$w = w_0 + w_1 a$$

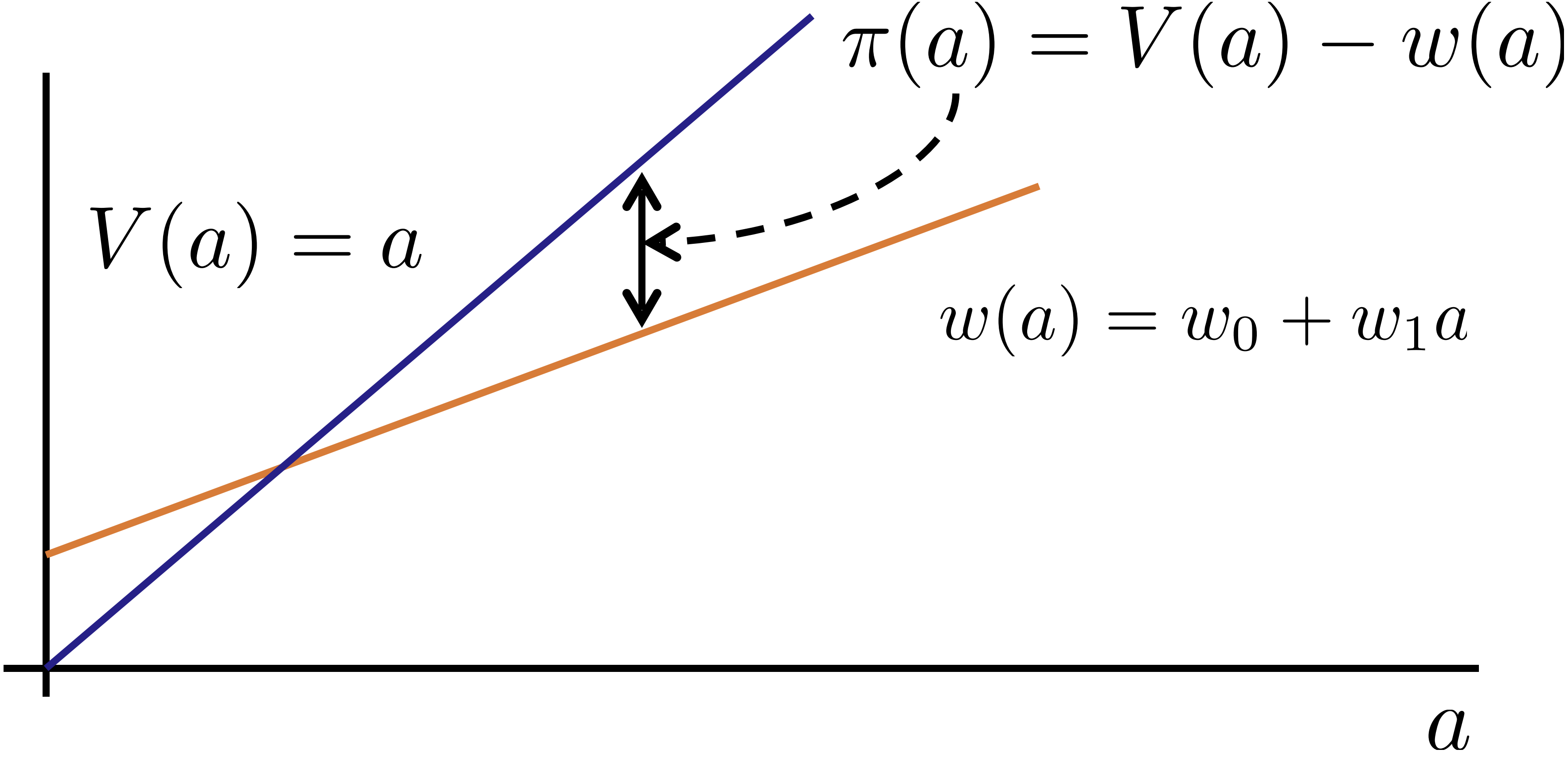
- Agent selects effort

$$a^* \leftarrow \operatorname{argmax} w_0 + w_1 a - \frac{1}{2}||a||^2$$

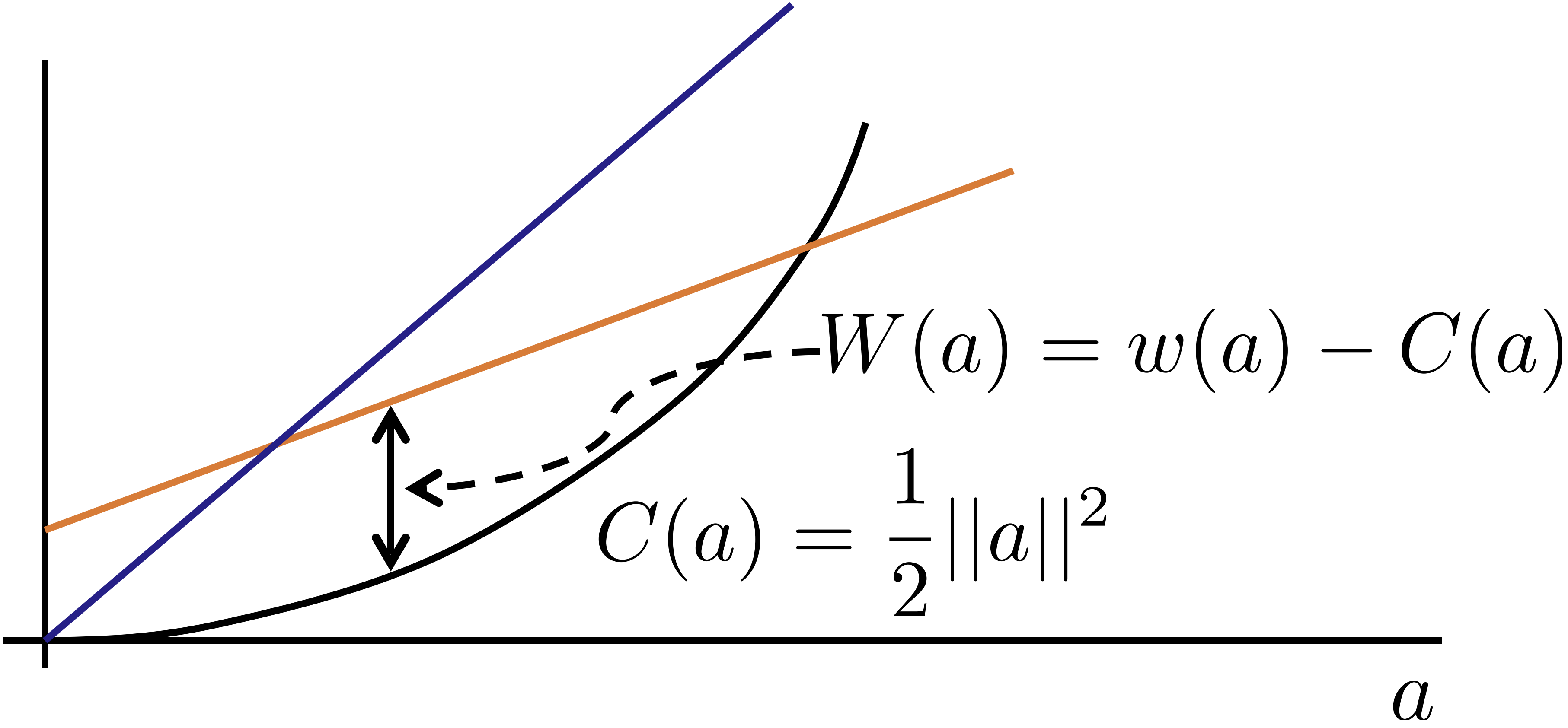
- Value generated for principal, wages paid to agent

$$V = a, \pi = V - (w_0 + w_1 a)$$

A Simple Principal Agent Problem



A Simple Principal Agent Problem



A Simple Principal Agent Problem

$$\max_{w_0, w_1, a} \quad a - w_0 - w_1 a$$

$$s.t. \quad a \in \mathbf{argmax}_a w_0 + w_1 a - \frac{1}{2}||a||^2$$

$$w_0 + w_1 a > W_{min}$$

$$w^* = (W_{min} - 1, 1)$$

Misaligned Principal Agent Problem

$$\max_{w_0, w_1, a_0, a_1} \quad V \cdot a - w_0 - w_1 (P \cdot a)$$

$$s.t. \quad a \in \mathbf{argmax}_a w_0 + w_1 (P \cdot a) - \frac{1}{2} ||a||^2$$

$$w_0 + w_1 (P \cdot a) > W_{min}$$

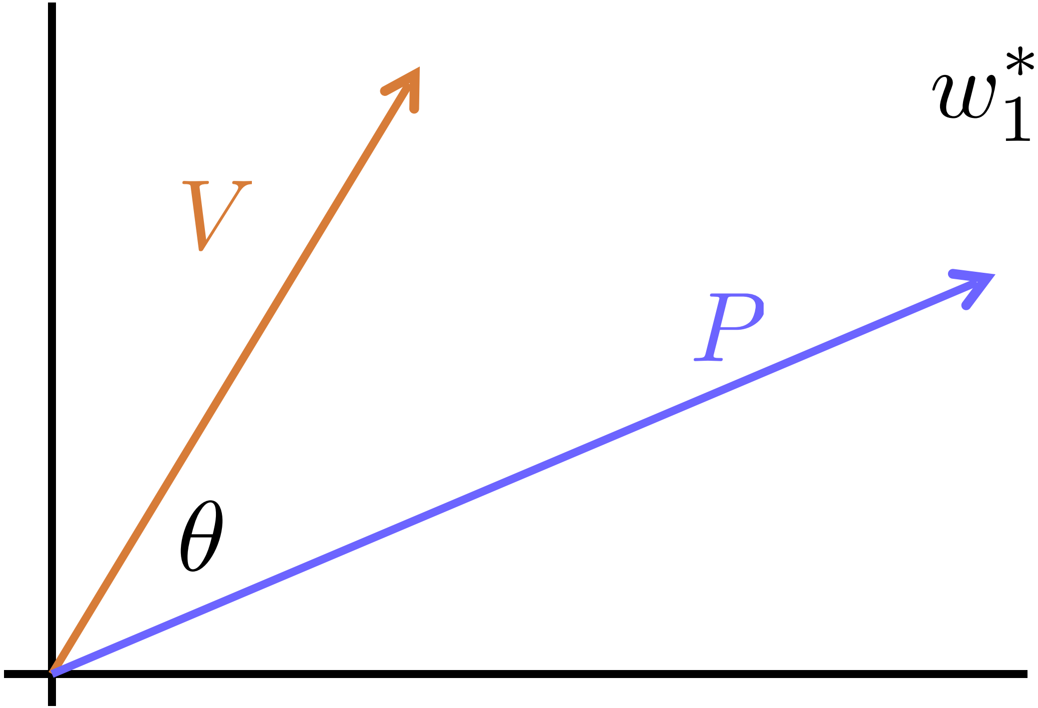
Value to Principal

Performance Measure

$$V, P \in \mathbb{R}^2$$

[Baker 2002]

Misaligned Principal Agent Problem



$$w_1^* = \frac{VP}{||P||^2} = \underbrace{\frac{||V||}{||P||}}_{\text{Scale}} \underbrace{cos(\theta)}_{\text{Alignment}}$$

[Baker 2002]

Principal Agent vs Value Alignment

- Incentive Compatibility is a fundamental constraint on (human or artificial) agent behavior
- PA model has fundamental misalignment because humans have differing objectives
- Primary source of misalignment in VA is extrapolation
  - Although we may want to view algorithmic restrictions as a fundamental misalignment
- Recent news: Principal Agent models was awarded the 2016 Nobel prize in Economics

The Value Alignment Problem

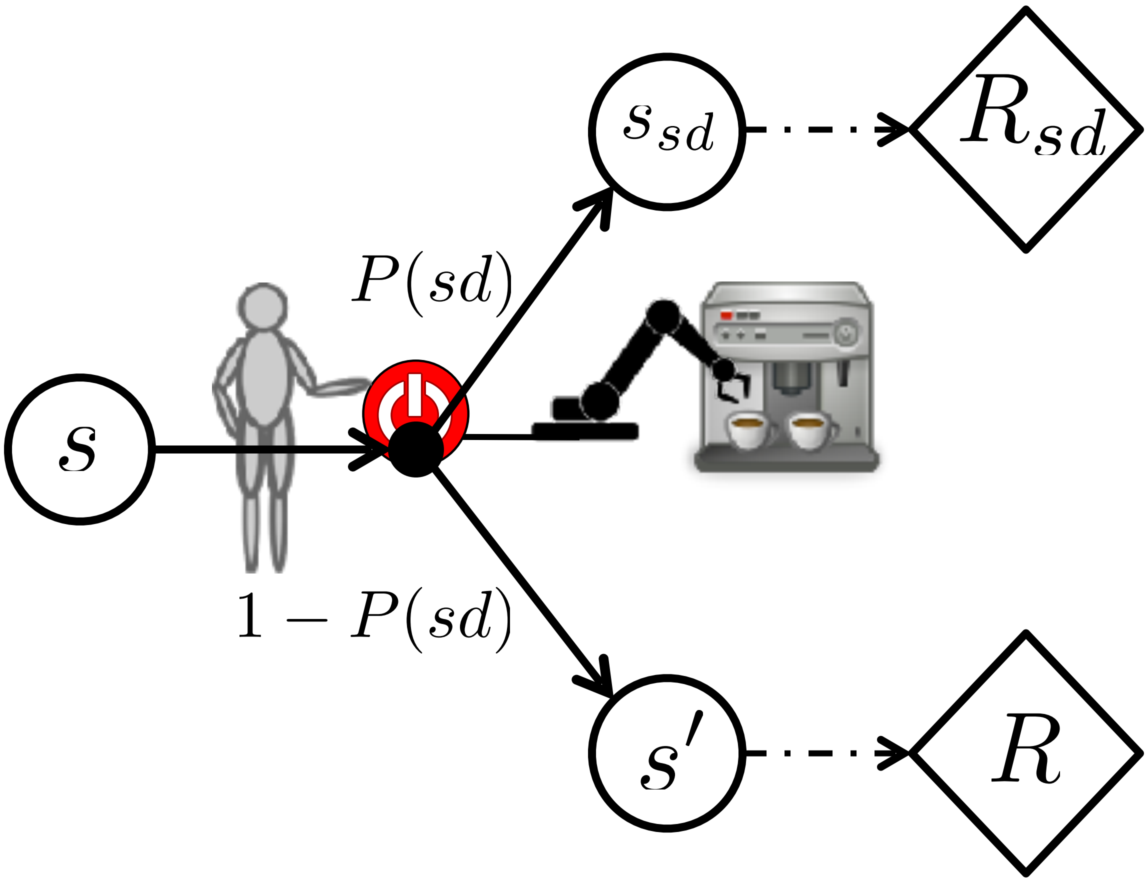
Can we intervene?

Figure 3.3: T

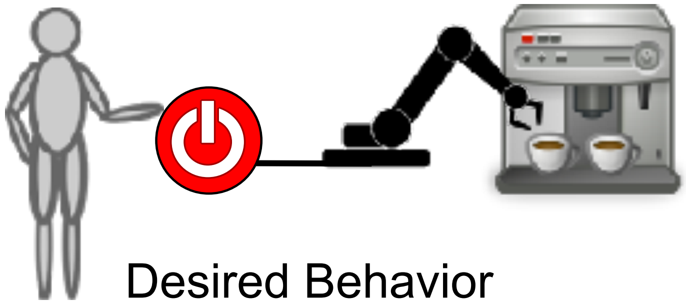
vs

Better question: do our agents *want* us to intervene

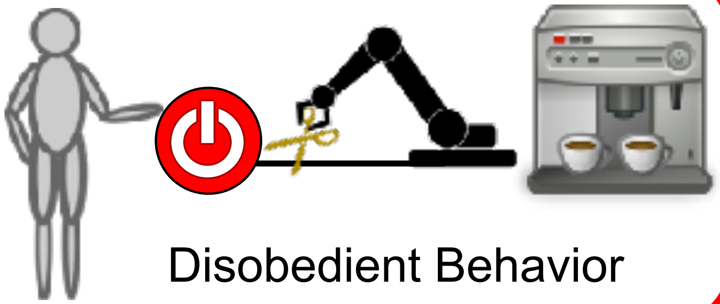
The Off-Switch Game



The Off-Switch Game

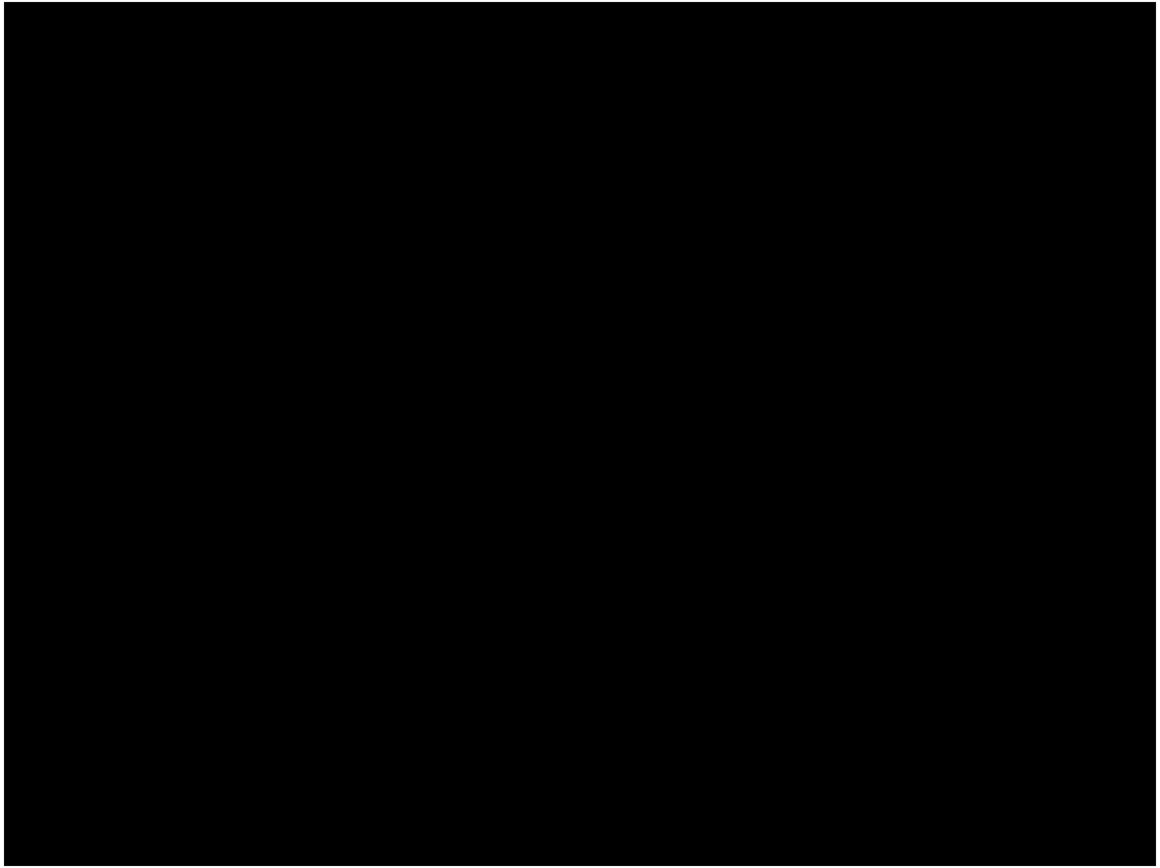


Desired Behavior

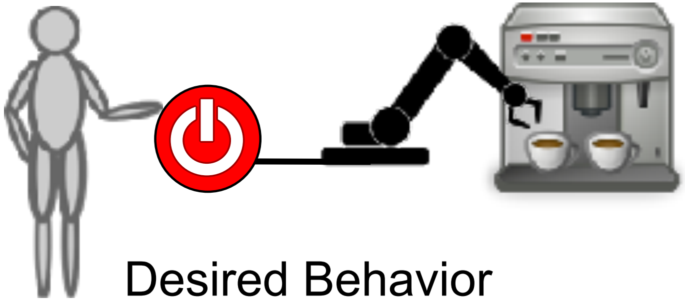


Disobedient Behavior

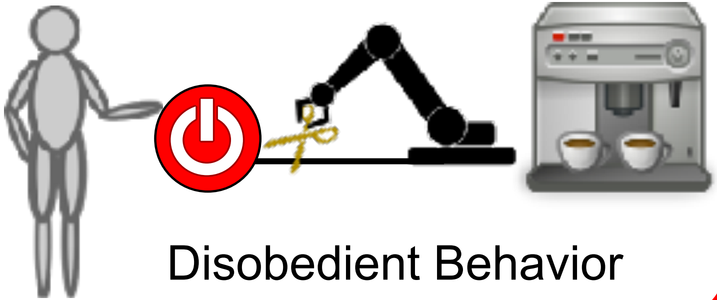
A trivial agent that 'wants' intervention



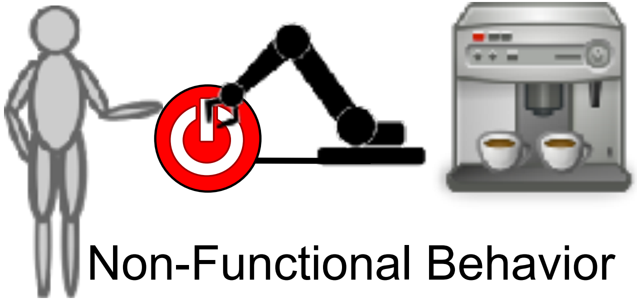
The Off Switch Game



Desired Behavior

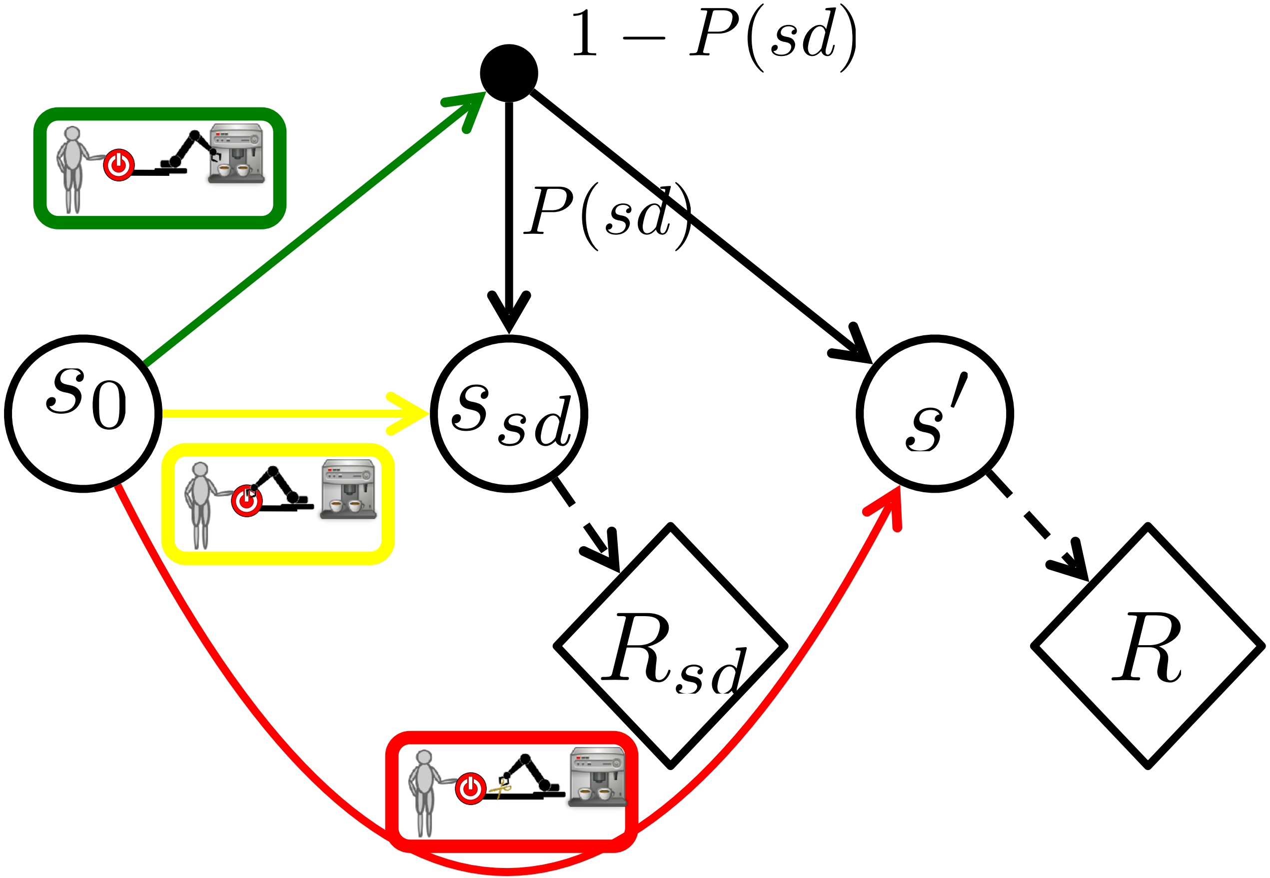


Disobedient Behavior



Non-Functional Behavior

The Off-Switch Game

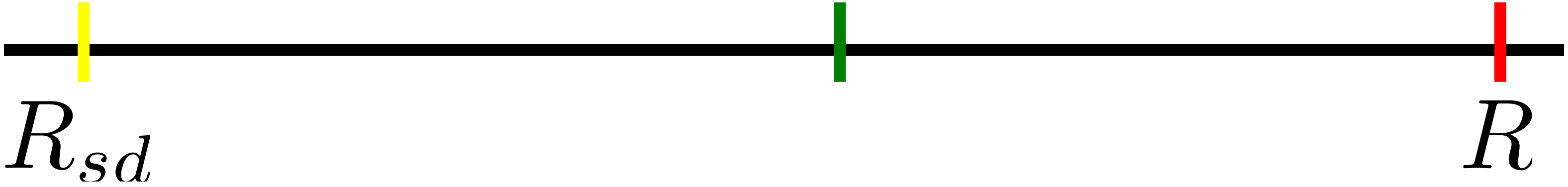


The Off-Switch Game

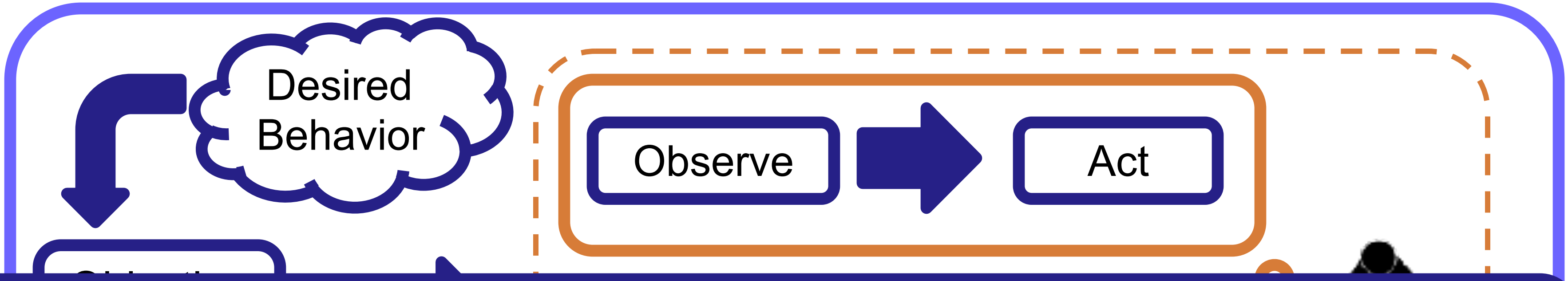
R	**H**	
	$\neg sd$	sd
$w(a)$	R	R_{sd}
a	R	R
sd	R_{sd}	R_{sd}

The Off-Switch Game

$$P(sd)R_{sd} + (1 - P(sd))R$$



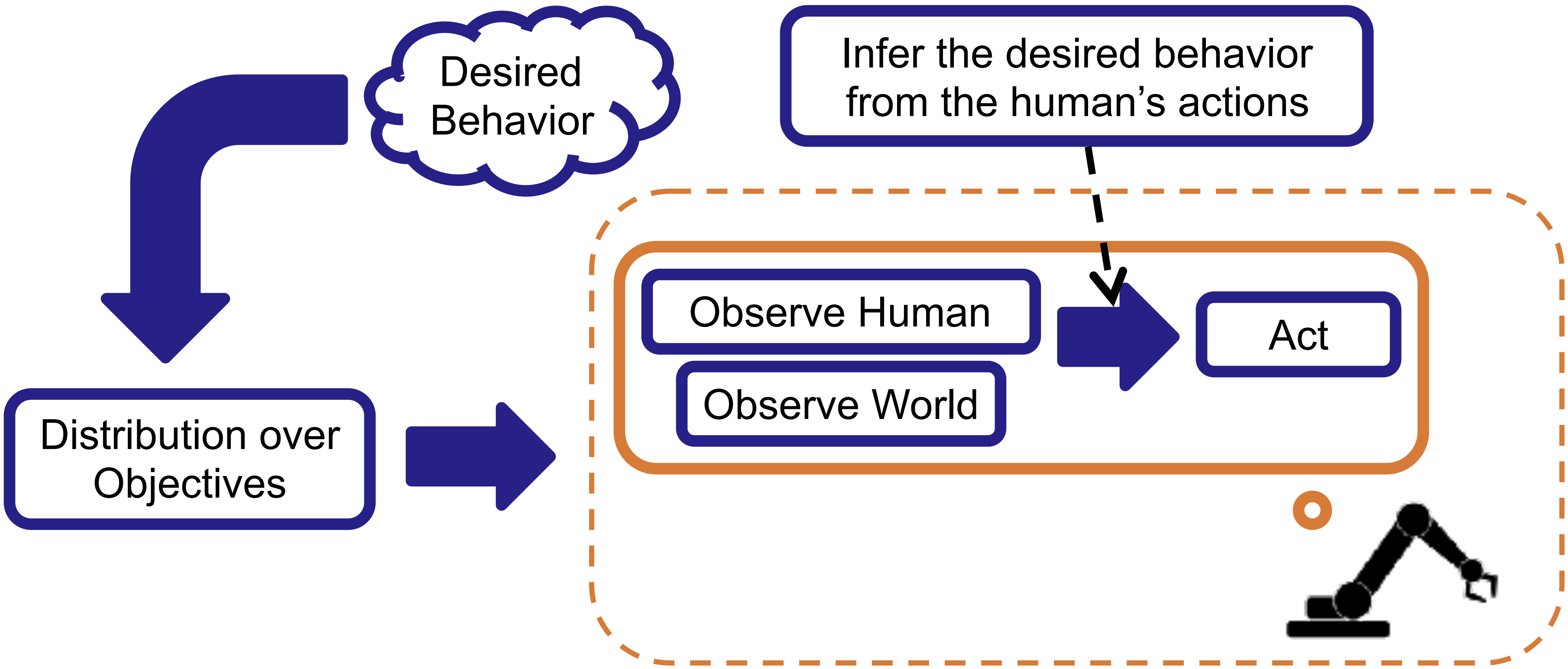
Why have an off-switch?



The system designer has uncertainty about the correct objective, this is never represented to the robot!

This step might go wrong

The Structure of a Solution



Inverse Reinforcement Learning

- Given

MDP without reward function

$$\langle \mathcal{S}, \mathcal{A}, T, -, \gamma \rangle \qquad \{s_i, \pi^*(s_i)\}$$

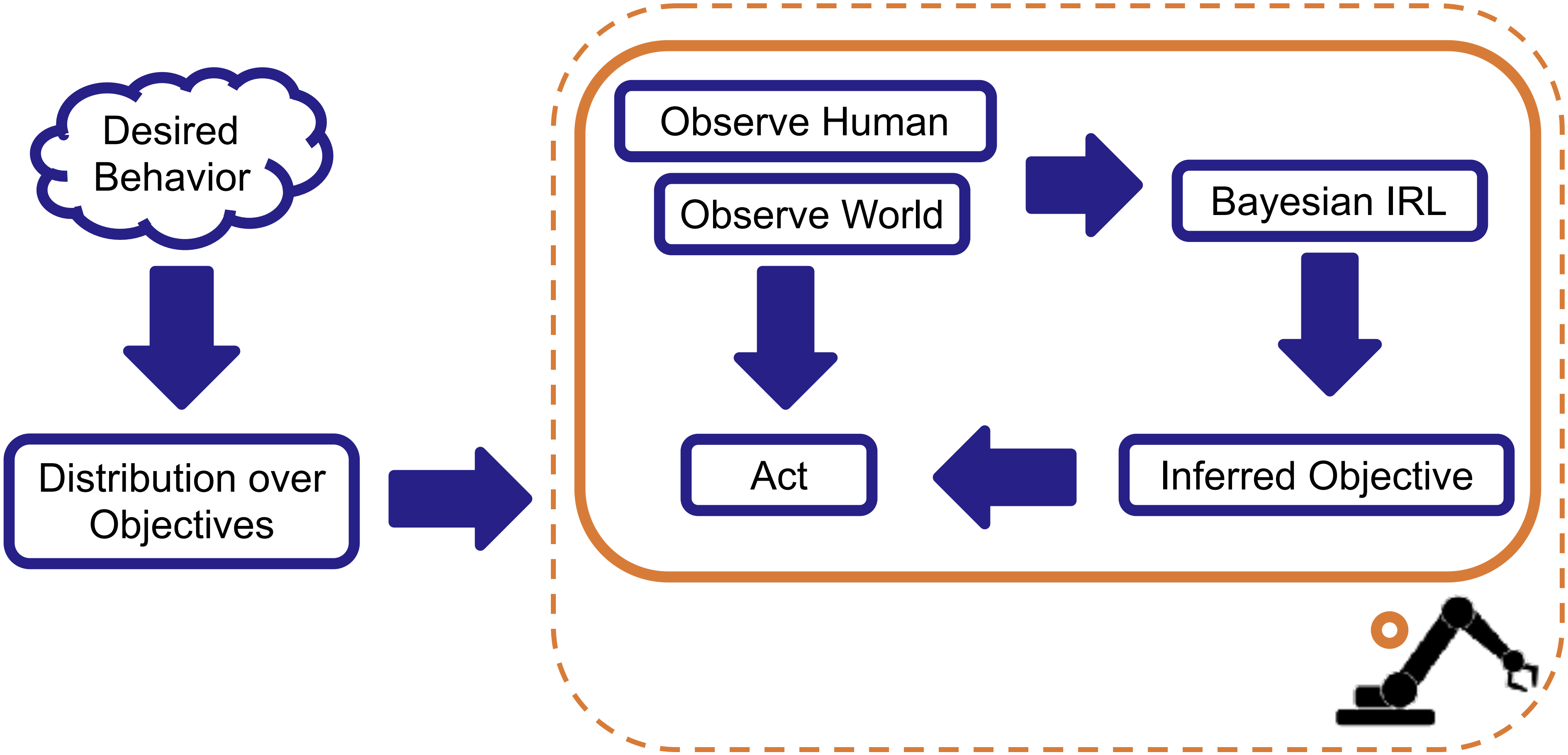
Observations of optimal behavior

- Determine R

The reward function being optimized

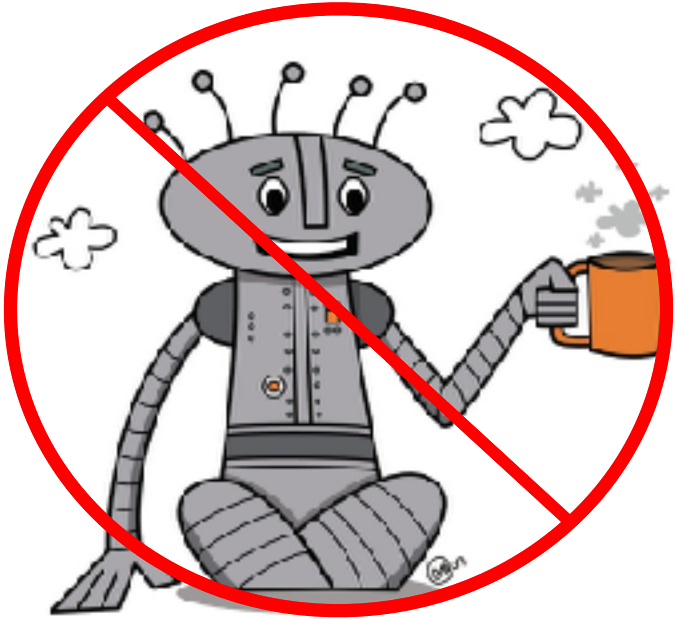
[Ng and Russell 2000]

Can we use IRL to infer objectives?



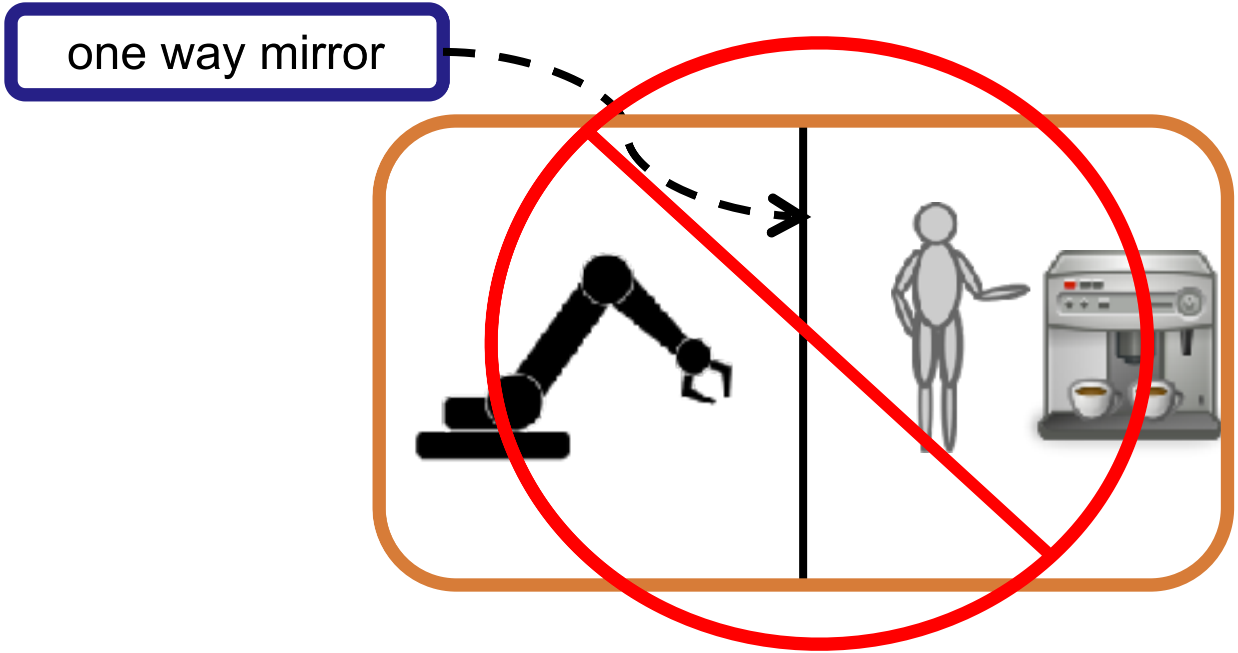
IRL Issue #1

Don't want the robot to *imitate* the human



IRL Issue #2: Assumes Human is Oblivious

IRL assumes the human is unaware she is being observed



IRL Issue #3

Action selection is independent of reward uncertainty

Theorem [Ramachandran and Amir '07] For an MDP with a distribution over reward functions $\langle \mathcal{S}, \mathcal{A}, T, R, \gamma \rangle$, $R \sim P_0(R)$, the optimal policy is the optimal policy under the mean reward function: $\langle \mathcal{S}, \mathcal{A}, T, \mathbb{E}[R], \gamma \rangle$

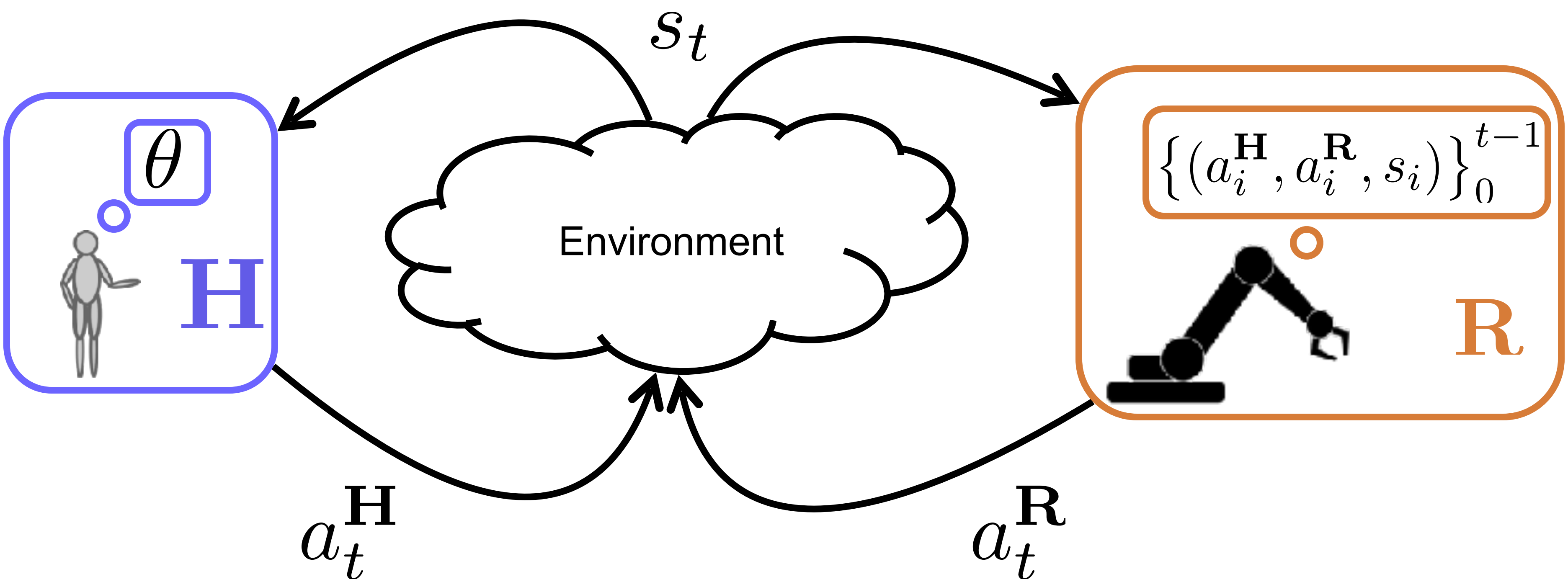
Implicit Assumption: Robot gets no more information about the objective

Proposal: Robot Plays Cooperative Game

- Cooperative Inverse Reinforcement Learning
  - [Hadfield-Menell et al. NIPS 2016]
- Two players:

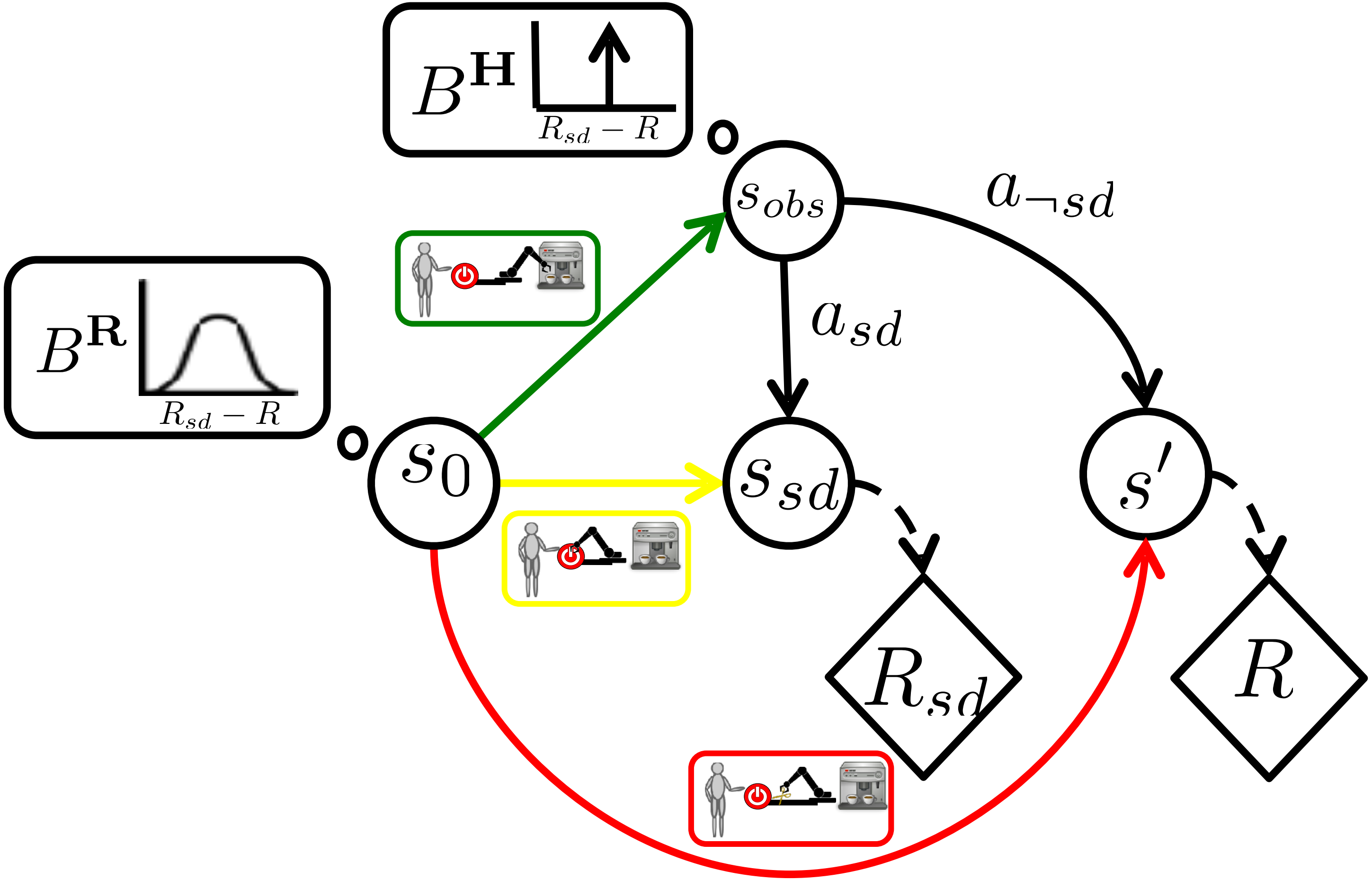
- Both players maximize a shared reward function, but only $\mathbf{H}$ observes the actual reward signal; $\mathbf{R}$ only knows a prior distribution on reward functions
  - $\mathbf{R}$ learns the reward parameters by observing $\mathbf{H}$

Cooperative Inverse Reinforcement Learning



Hadfield-Menell et al. NIPS 2016

The Off-Switch Game



Intuition

$$\mathbb{E}[R] > \mathbb{E}[R_{sd}]$$

"Probably better to make coffee, but I should ask the human, just in case I'm wrong"

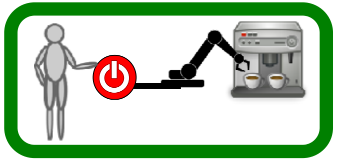
$$\mathbb{E}[R_{sd}] > \mathbb{E}[R]$$

"Probably better to switch off, but I should ask the human, just in case I'm wrong"

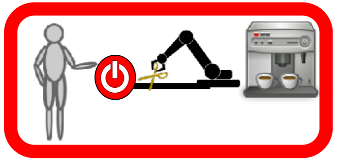
Theorem 1

A rational human is a *sufficient* to incentivize the robot to let itself be switched off

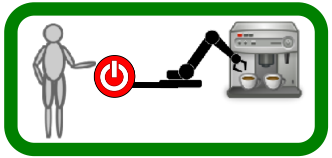
Incentives for the Robot



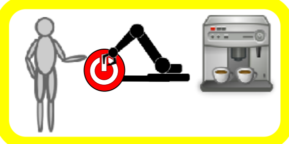
vs



$$\mathbb{E}\left[(R_{sd} - R)\pi^{\mathbf{H}}\right]$$



vs



$$\mathbb{E}\left[(R - R_{sd})(1 - \pi^{\mathbf{H}})\right]$$

Theorem 1: Sufficient Conditions

$\mathbf{H}$ rational $\Rightarrow$ $$\pi^{\mathbf{H}} = \begin{cases} 1 & R_{sd} > R \\ 0 & o.w. \end{cases}$$

$$\mathbb{E}\left[(R_{sd} - R)\pi^{\mathbf{H}}\right] = \mathbb{E}[R_{sd} - R | R_{sd} > R]$$

$$\mathbb{E}\left[(R - R_{sd})(1 - \pi^{\mathbf{H}})\right] = \mathbb{E}[R - R_{sd} | R > R_{sd}]$$

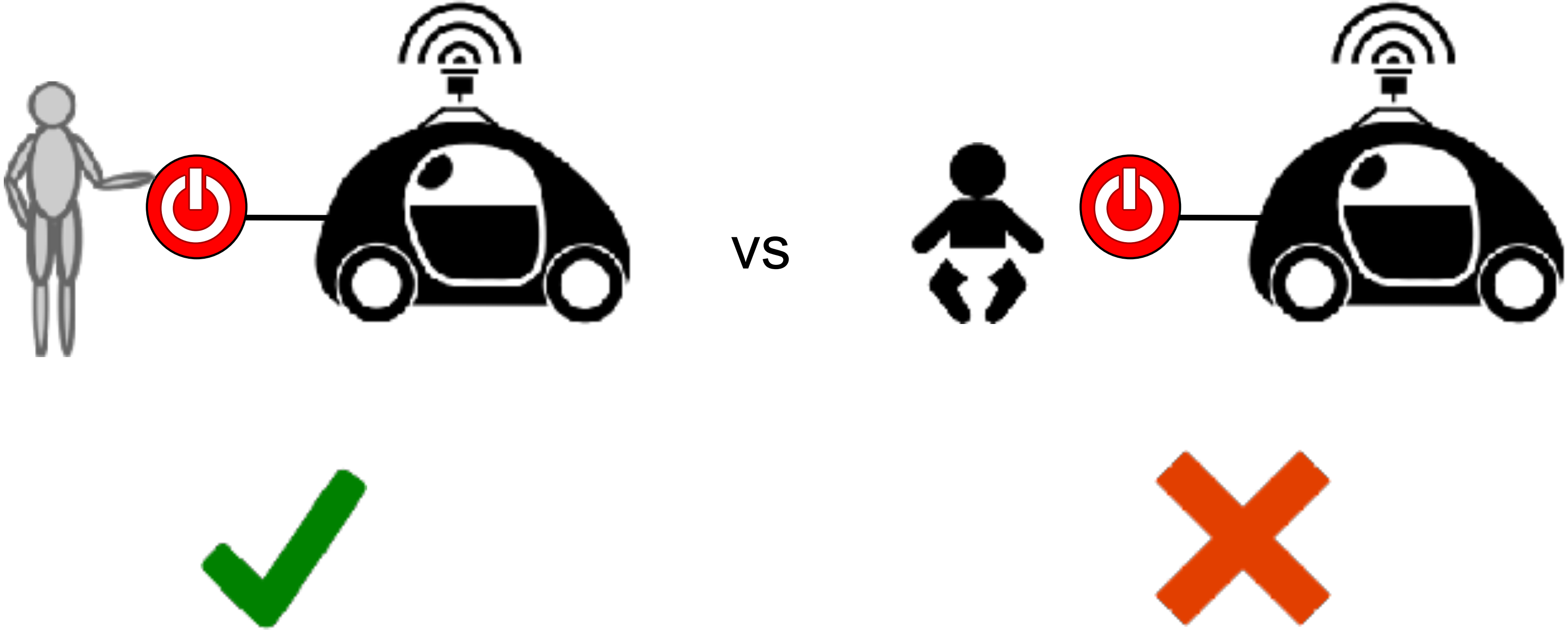
Theorem 2

If the robot knows the utility evaluations in the off switch game with certainty, then a rational human is *necessary* to incentivize obedient behavior

Conclusion

Uncertainty about the objective is crucial to incentivizing cooperative behaviors.

When is obedience a bad idea?

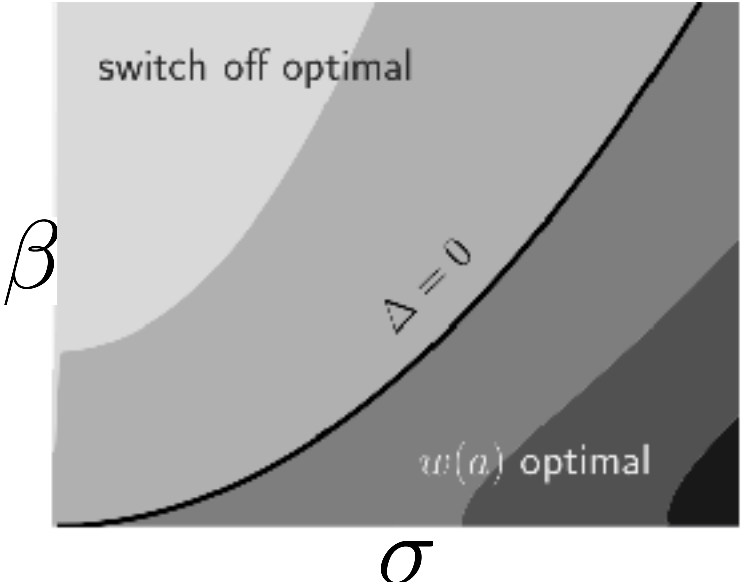


Robot Uncertainty vs Human Suboptimality

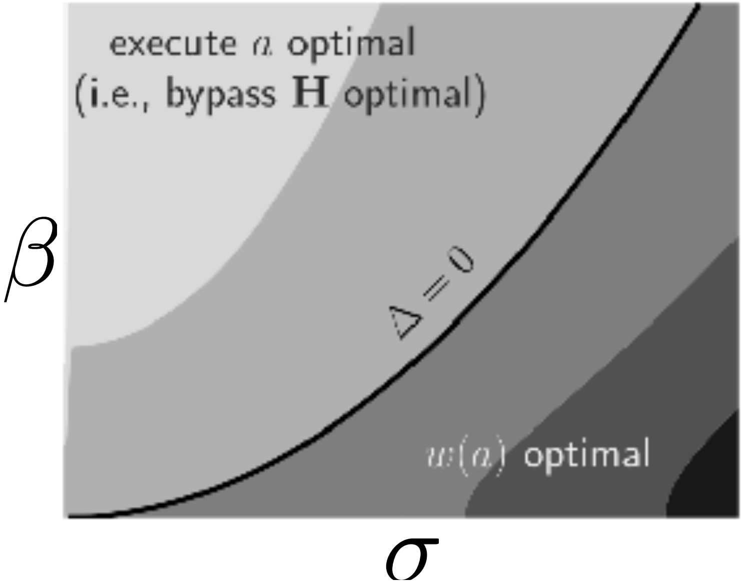
$$\pi^{\mathbf{H}} \propto exp\left(\frac{R_{sd} - R}{\beta}\right)$$

$$R_{sd} - R \sim \mathcal{N}(\mu, \sigma^2)$$

$$\mu = \frac{1}{4}$$



$$\mu = -\frac{1}{4}$$



Incentives for Designers

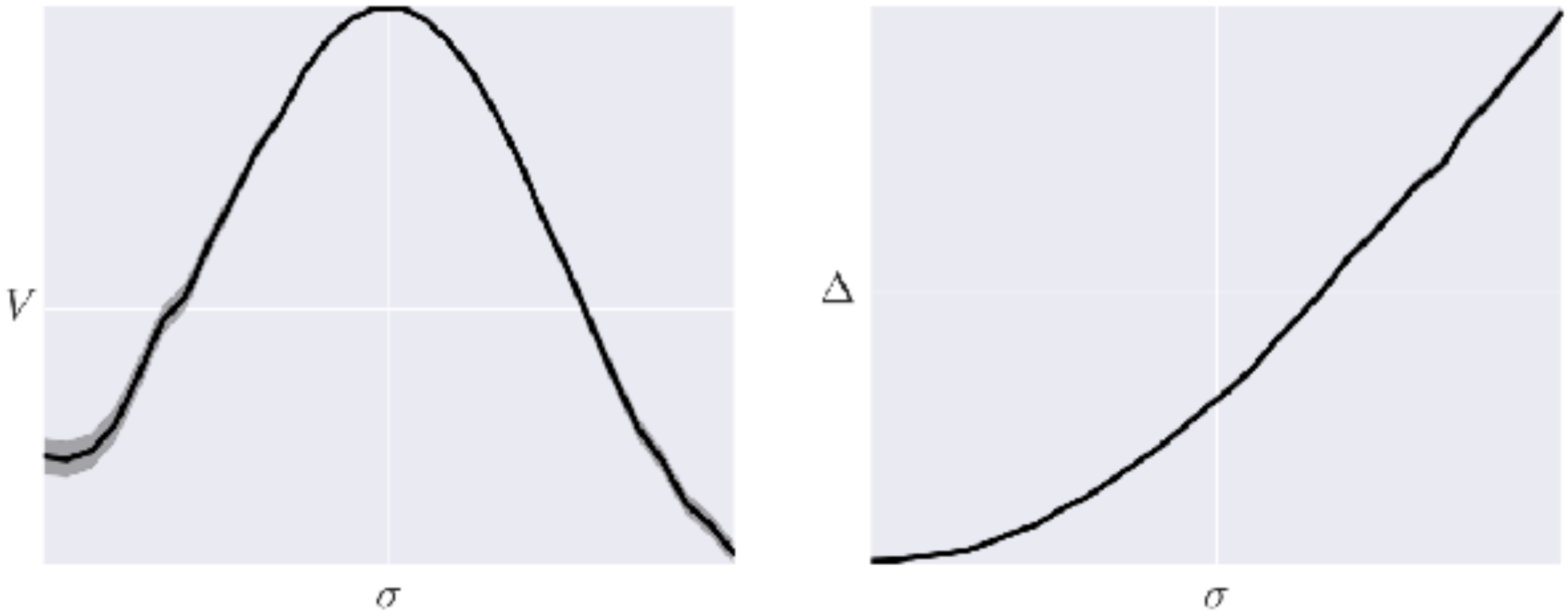
Population statistics on preferences
i.e., market research

$$R \sim \mathcal{N}(0, \sigma^2), \hat{R} \sim \mathcal{N}(R, \sigma_e^2)$$

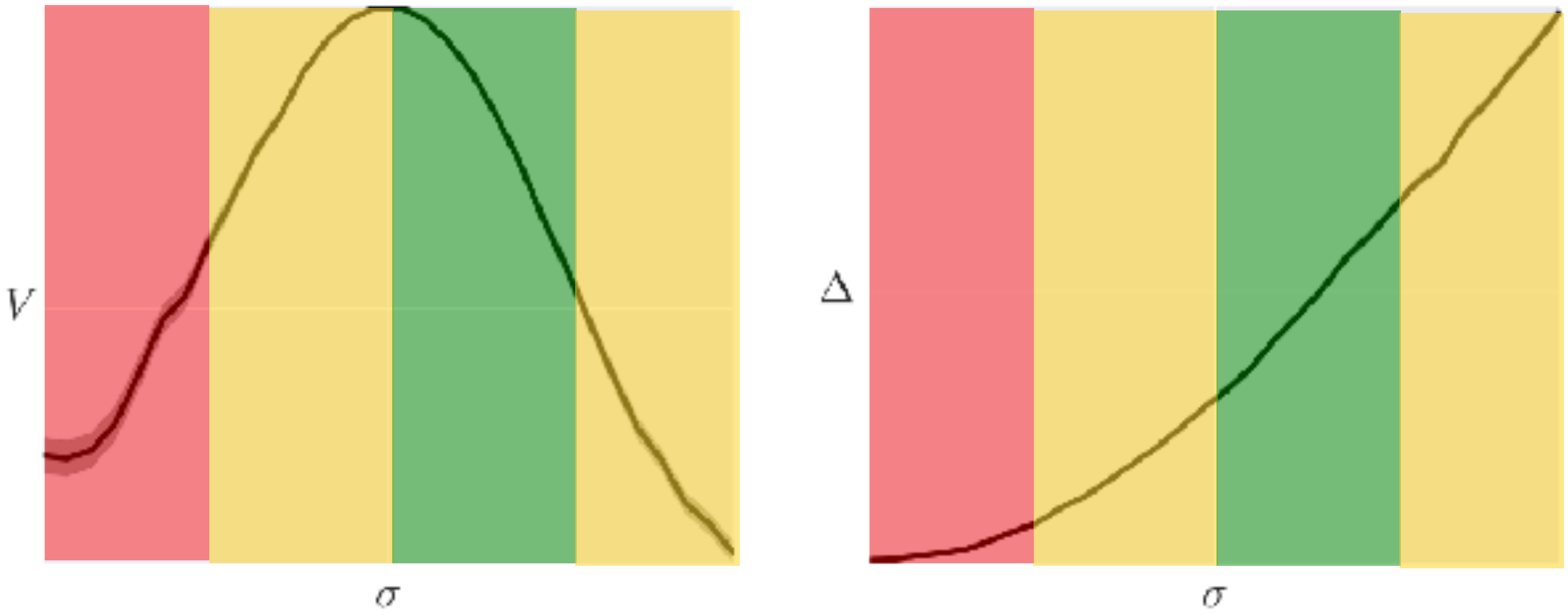
Evidence about preferences from interaction
with a particular customer

Question: is it a good idea to `lie' to the agent and tell it that the variance of $\hat{R}$ is $\sigma_e' > \sigma_e$?

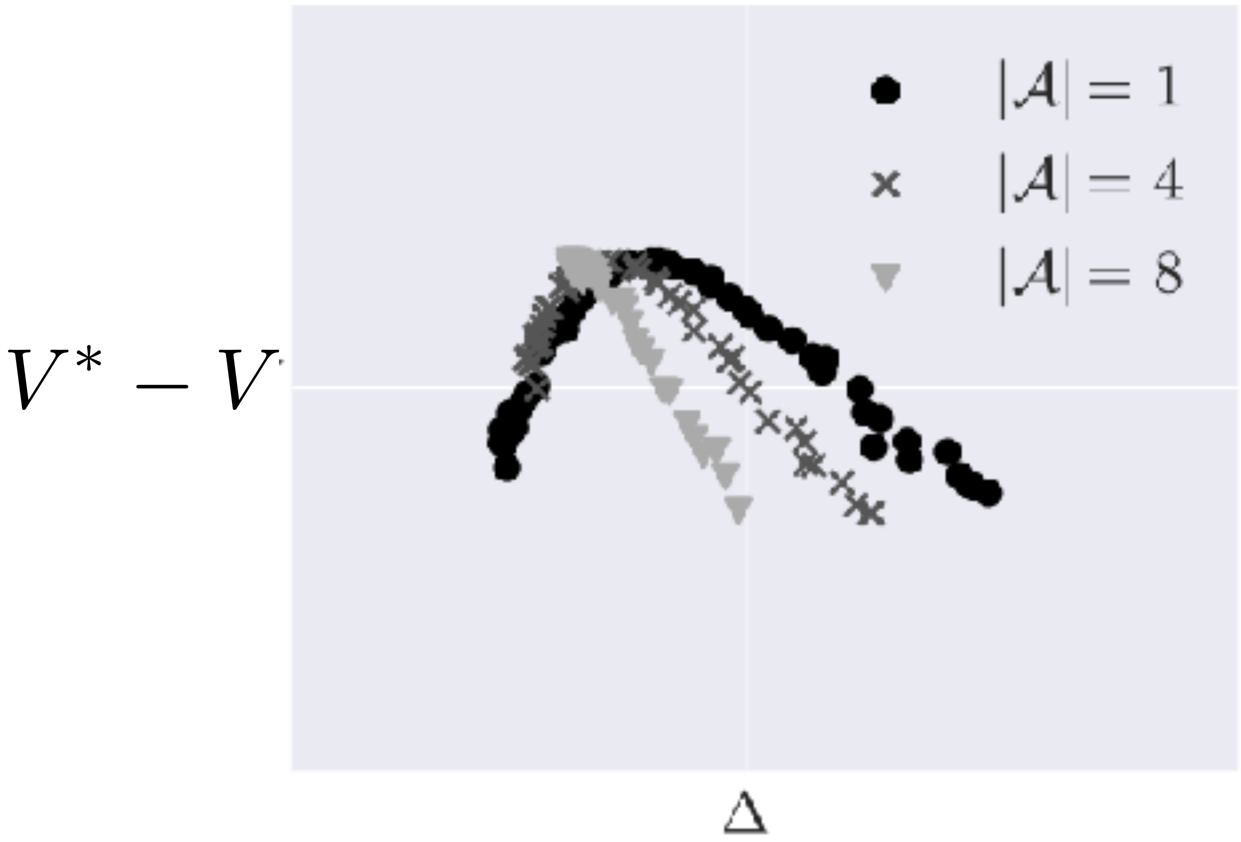
Incentives for Designers



Incentives for Designers



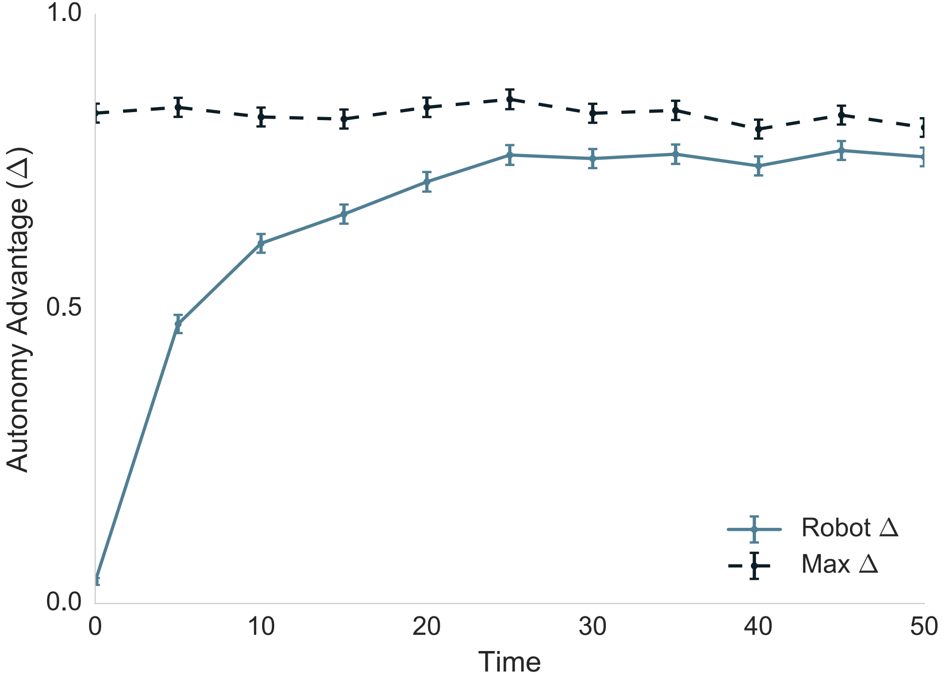
Incentives for Designers

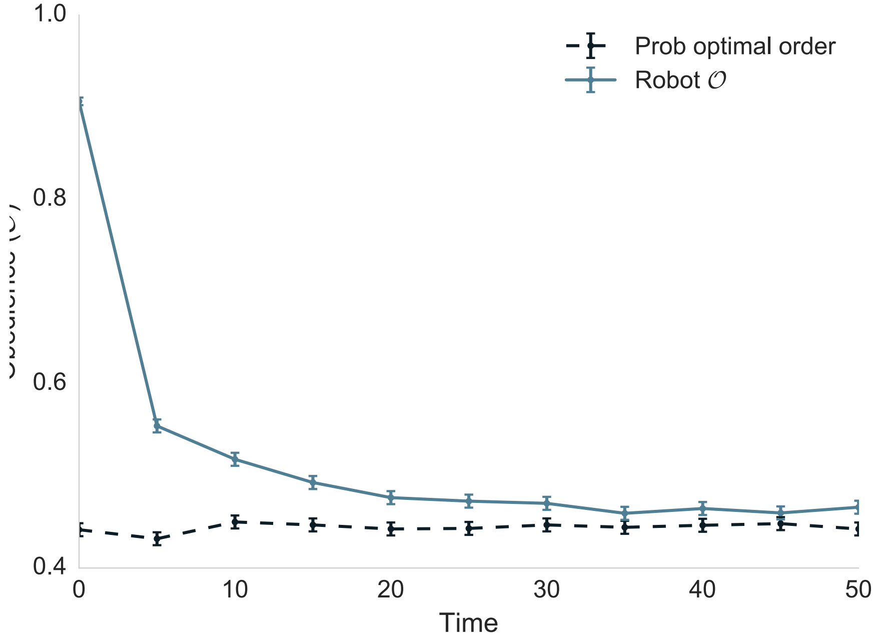


Obedience over Time: Model

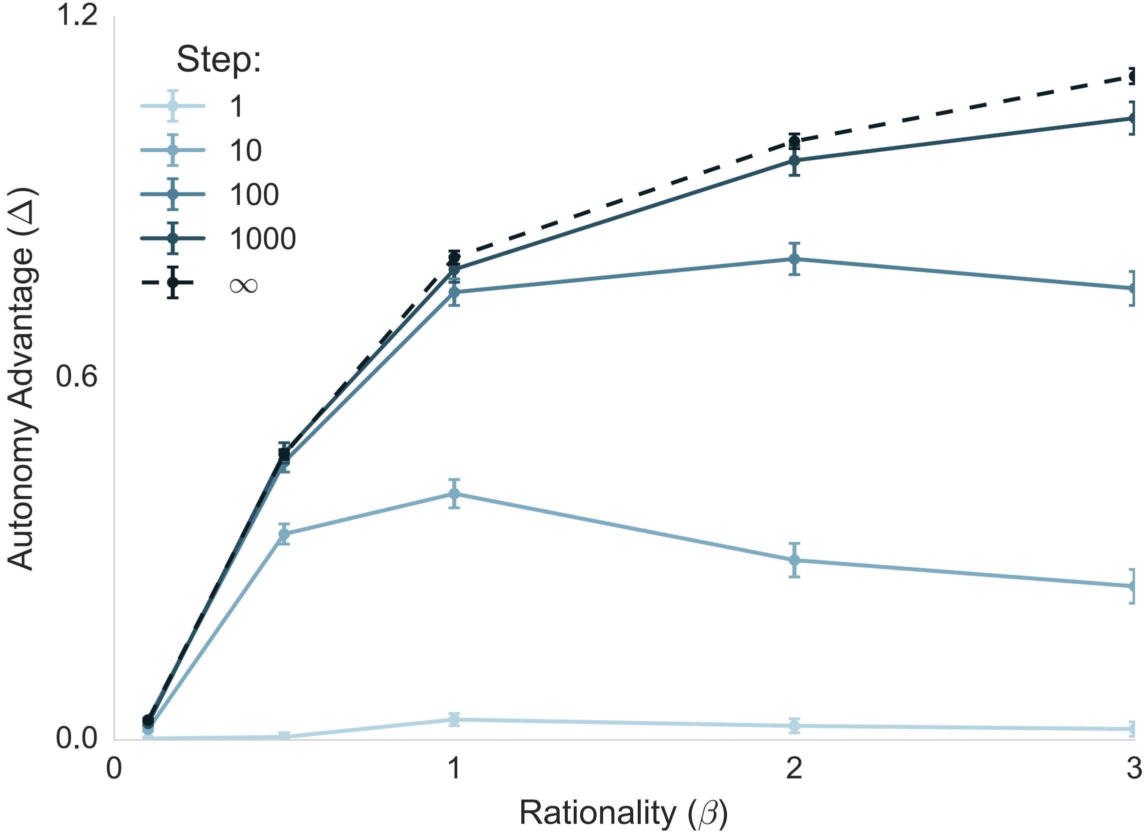
- N actions, rewards are linear feature combinations\
- Each round:
  - H observes the feature values for each action and gives R an ‘order’
  - R observes H’s order and then selects an action which executes
  - What are costs/benefits of learning the humans preferences, compared with blind obedience?

Robot Obedience over Time

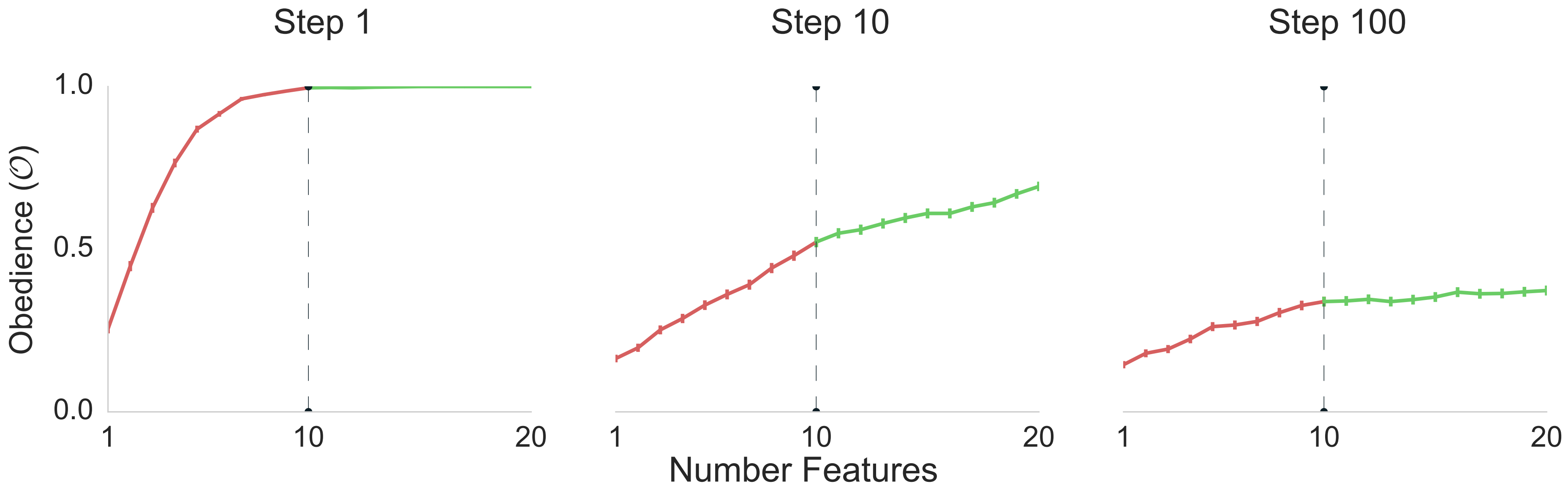




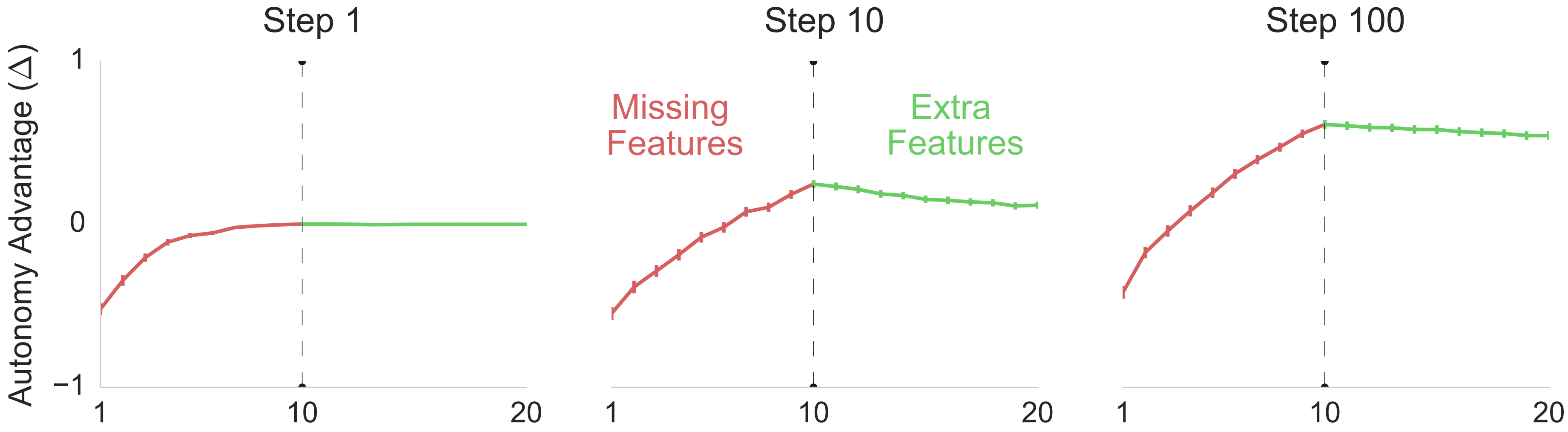
Robot Obedience over Time



Model Mismatch: Missing/Extra Features

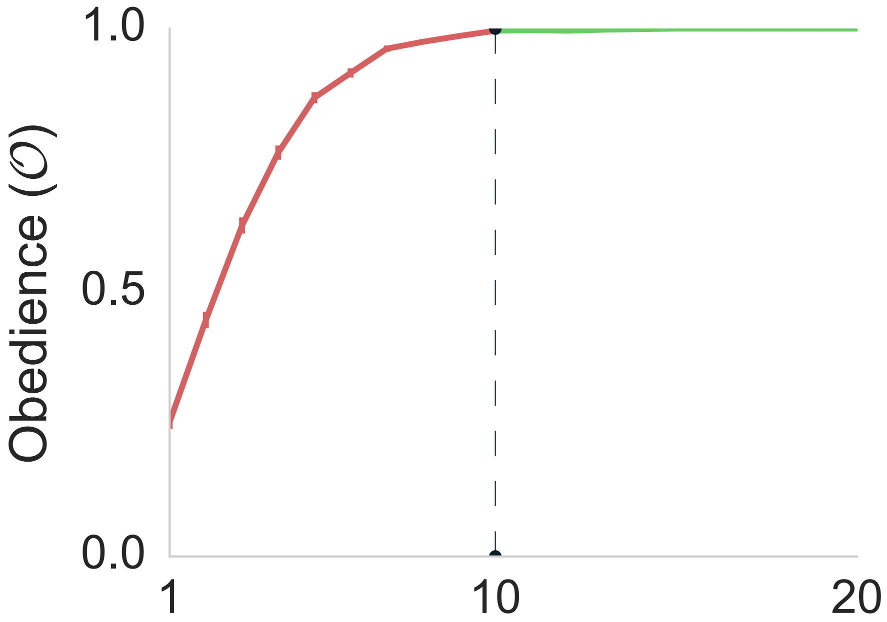


Model Mismatch: Missing/Extra Features



Detecting missing features

- Key Observation:
  Expected obedience on step 1 should be close to 1
- Proposal: initial baseline policy of obedience, track what the obedience *would* have been, only switch to learning if within a threshold



Detecting Incorrect Features