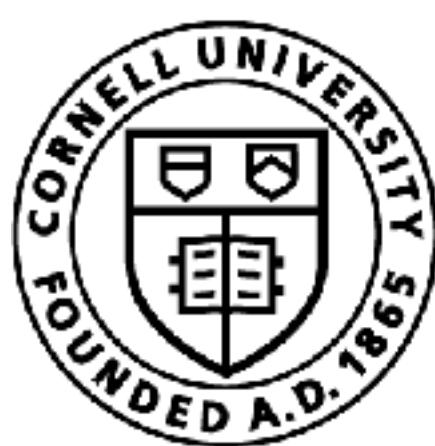


Behavior Cloning, Feedback and Covariate Shift

Sanjiban Choudhury



Cornell Bowers CIS
Computer Science

What have we learnt so far?

1. How do define a MDP
2. How to solve a MDP given I know S , A , C , T

But there are challenges in applying this

Q1. What if I can write down my costs, but my transitions are unknown?

Reinforcement Learning! (Later in the course)

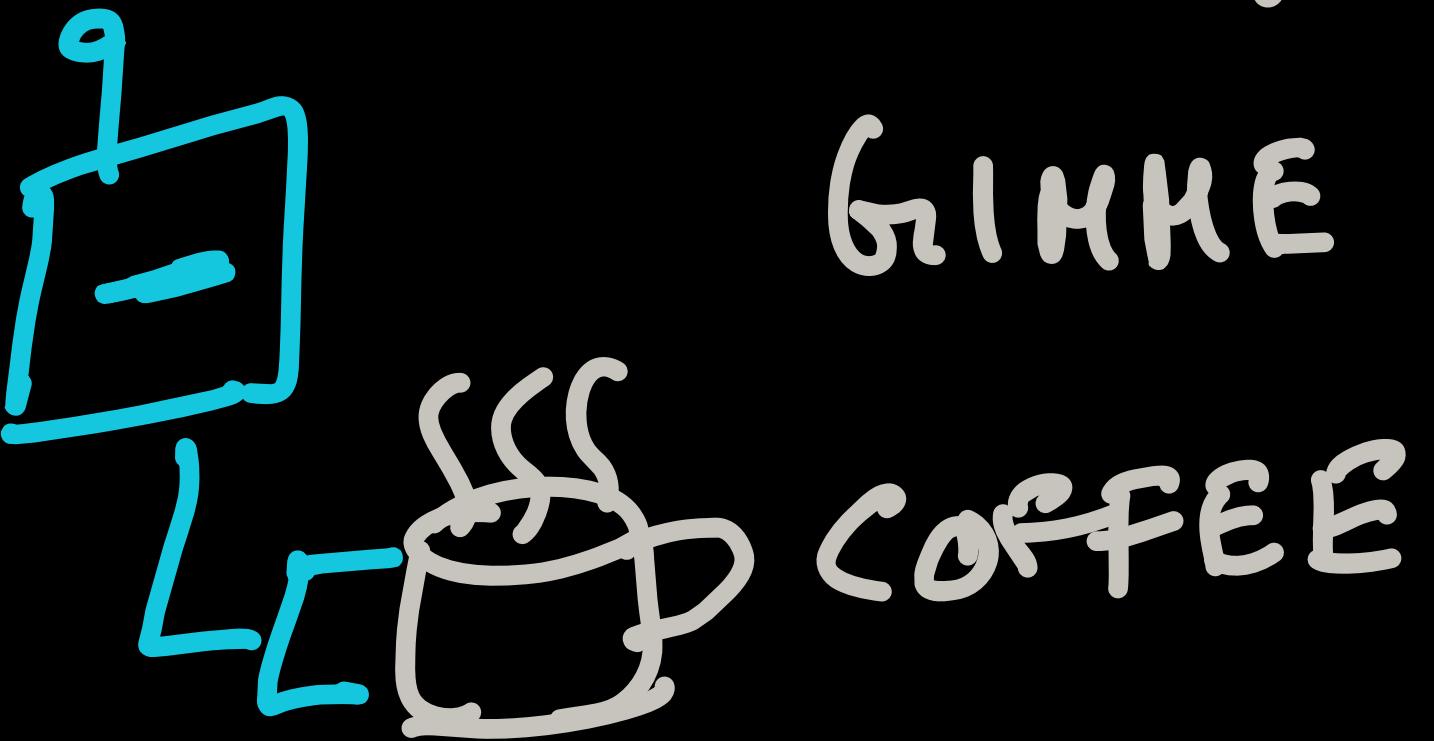
Q2. But what if even writing down costs is hard?

Imitation Learning! (Today)

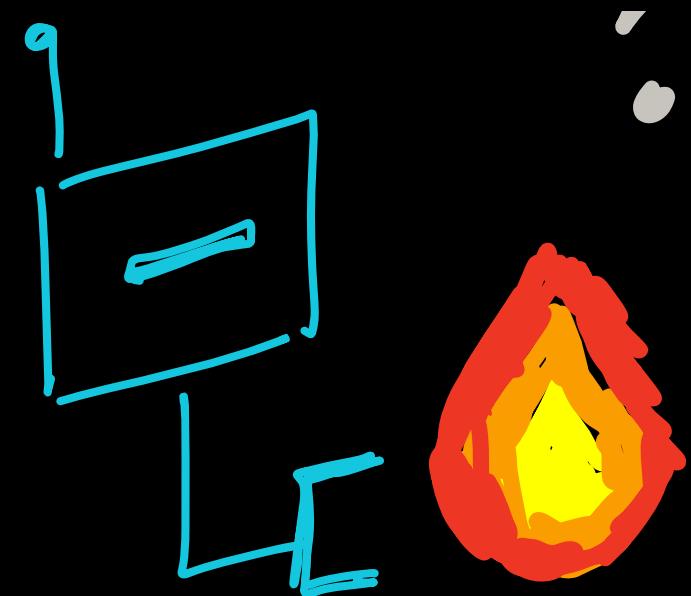
How do we program
robots to do tasks?

Programming a task ...

tell the robot to make coffee ..

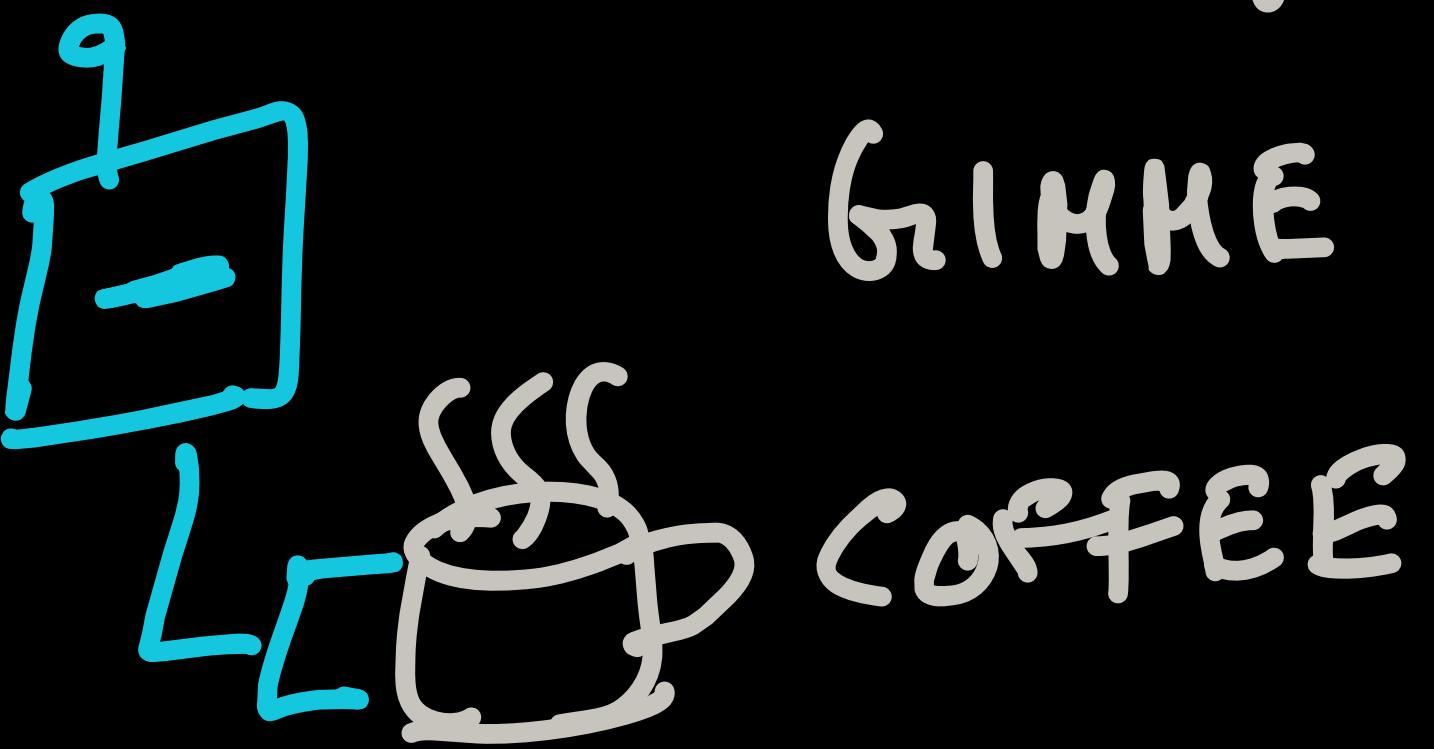


robot burns down
the house!



Programming a task ...

tell the robot to make coffee ..



DON'T ...
burn down the house
steal the neighbors coffee
don't make a mess

...



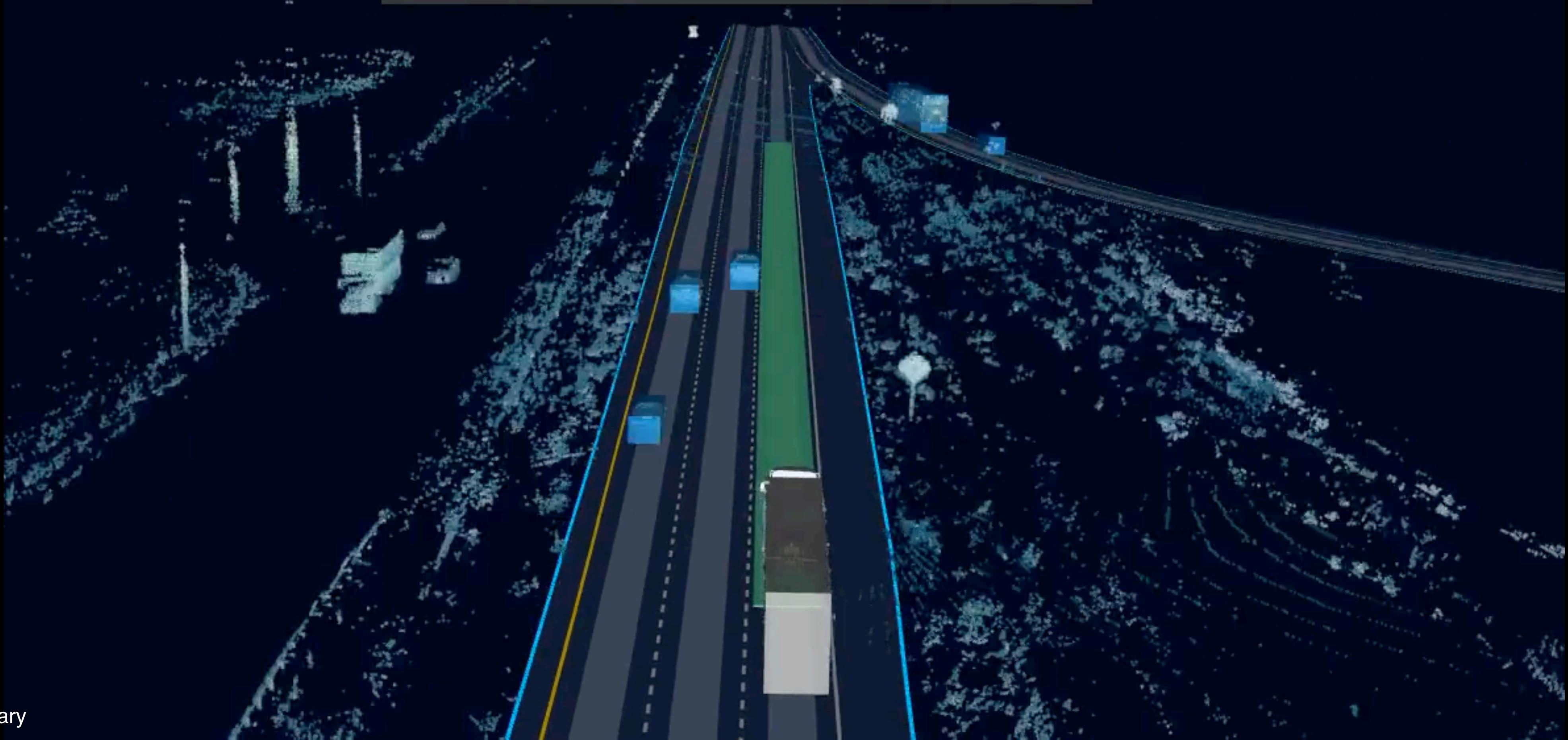
STRAIGHT
2.1 MI

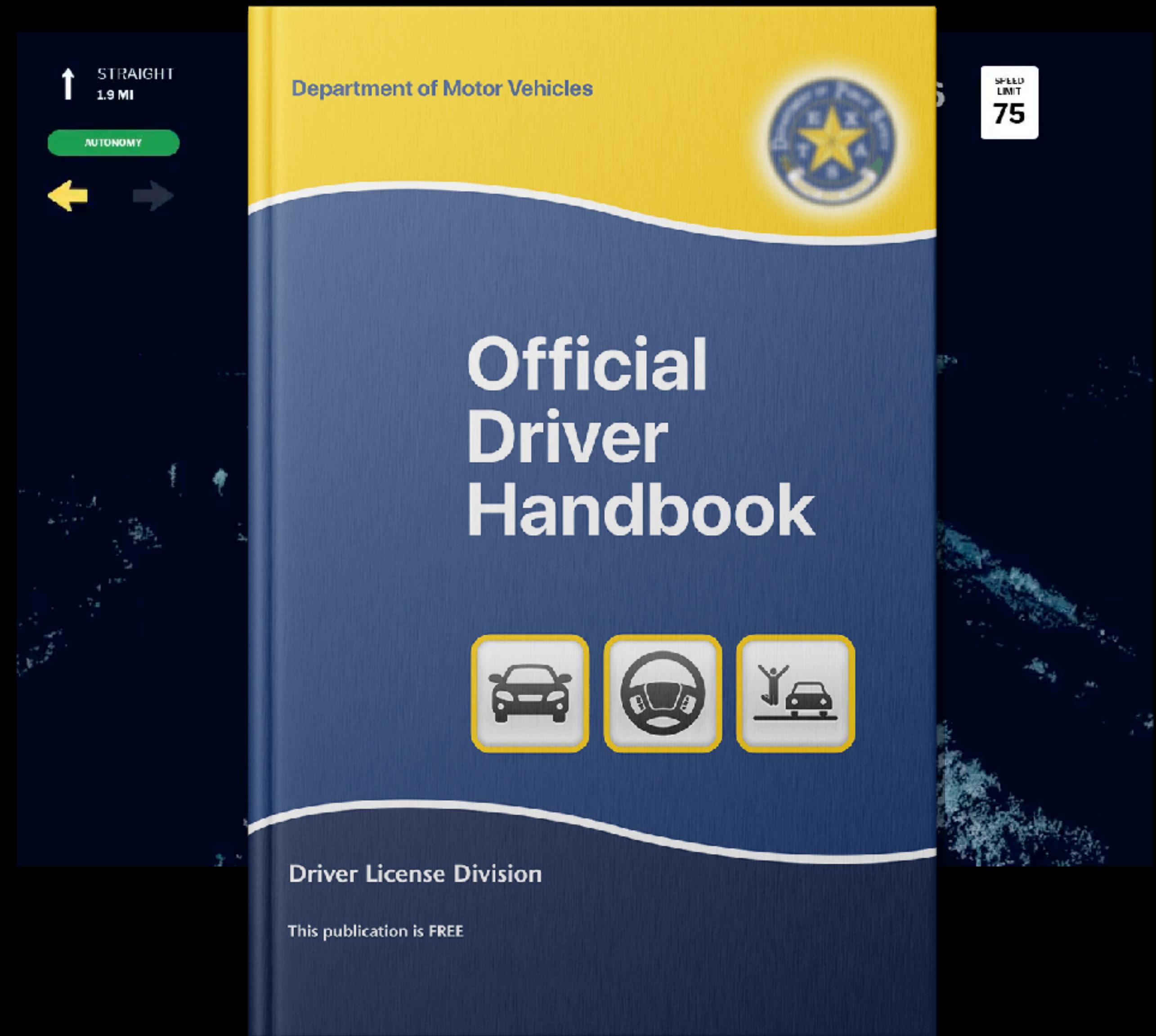
AUTONOMY



65
MPH

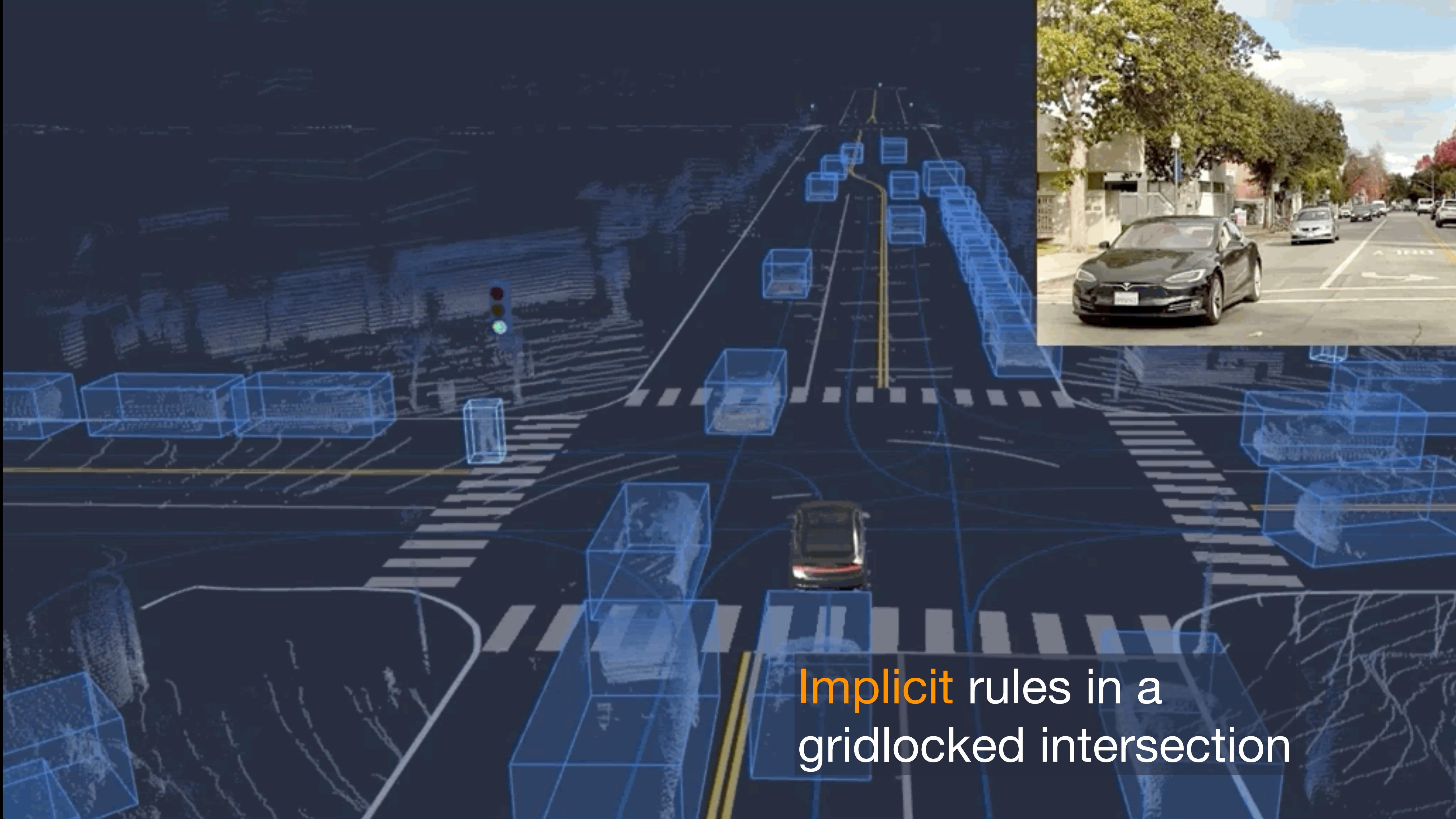
SPEED
LIMIT
75





The implicit rules of human driving





Implicit rules in a
gridlocked intersection



Explicitly programming
rules may be tedious ...

... but rules are implicit
in how we drive everyday!



Imitation Learning

Implicitly program robots

Activity!



Think-Pair-Share!

Think (30 sec): What are the various ways to give input to a robot to teach it a new task?

Pair: Find a partner

Share (45 sec): Partners exchange ideas



Imitation learning is everywhere

Helicopter Aerobatics



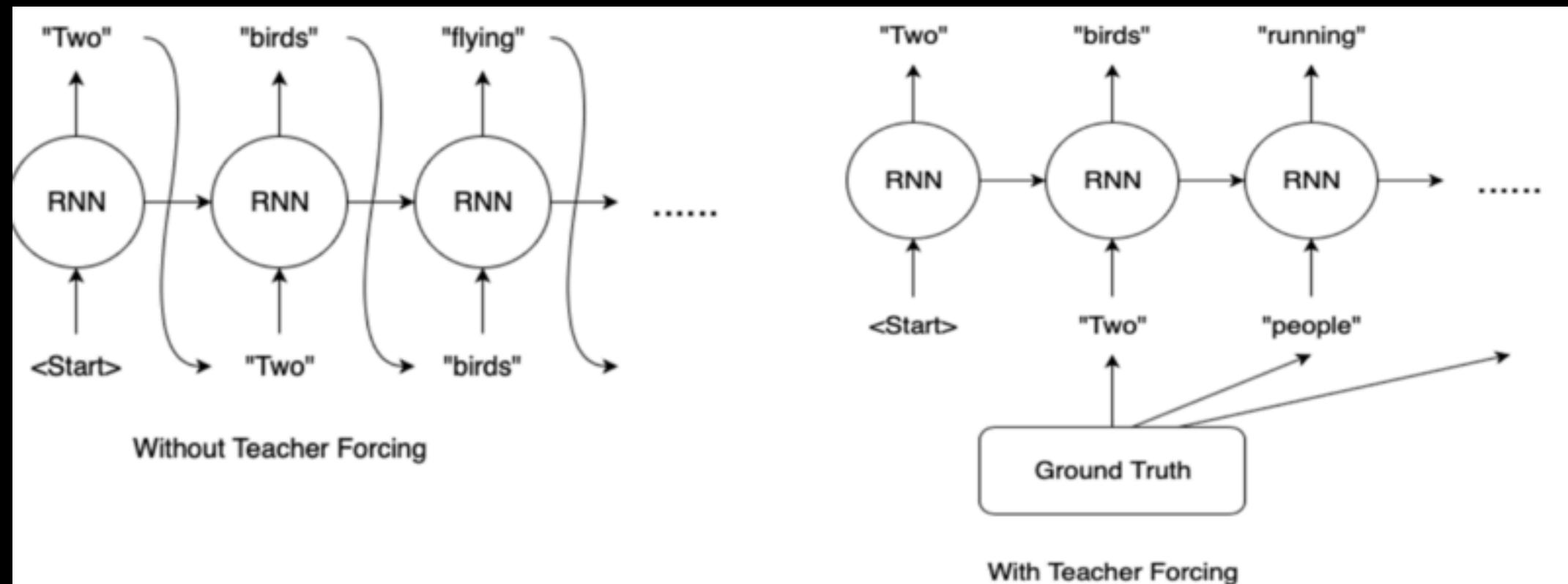
Abbeel et al. 2009



Game AI

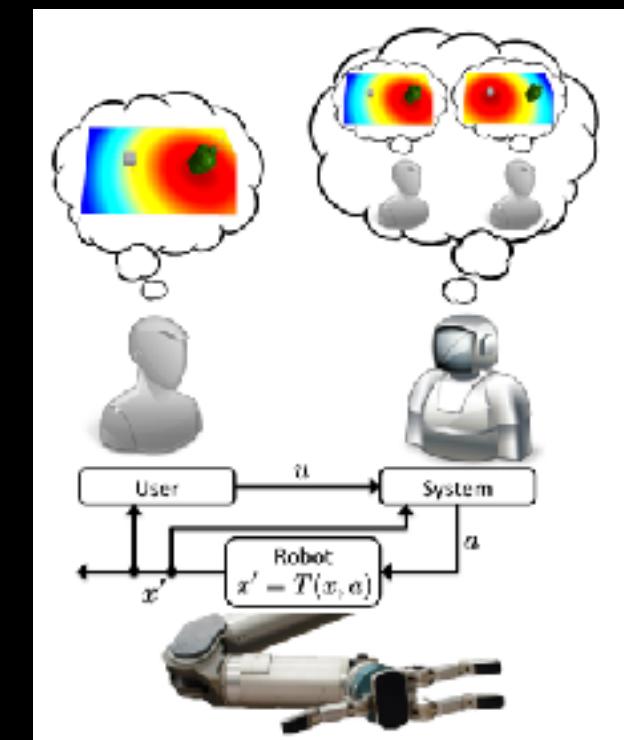
Kozik et al. 2021

Sequence models in NLP



Daume et al. 2009

Shared autonomy



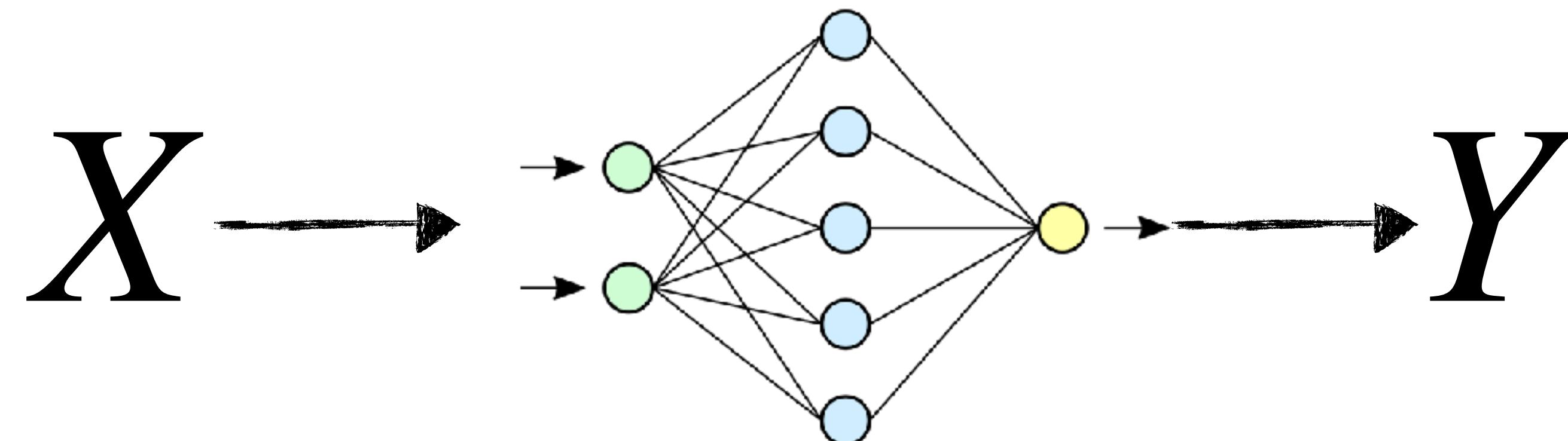
Javdani et al. 2015

How do we solve imitation learning?

Treat robotics as a “simple” ML problem ...



Ultimately, we just need to learn a function



Input
(State)

Model
(Policy)

Output
(Action)

Behavior Cloning

Behavior Cloning

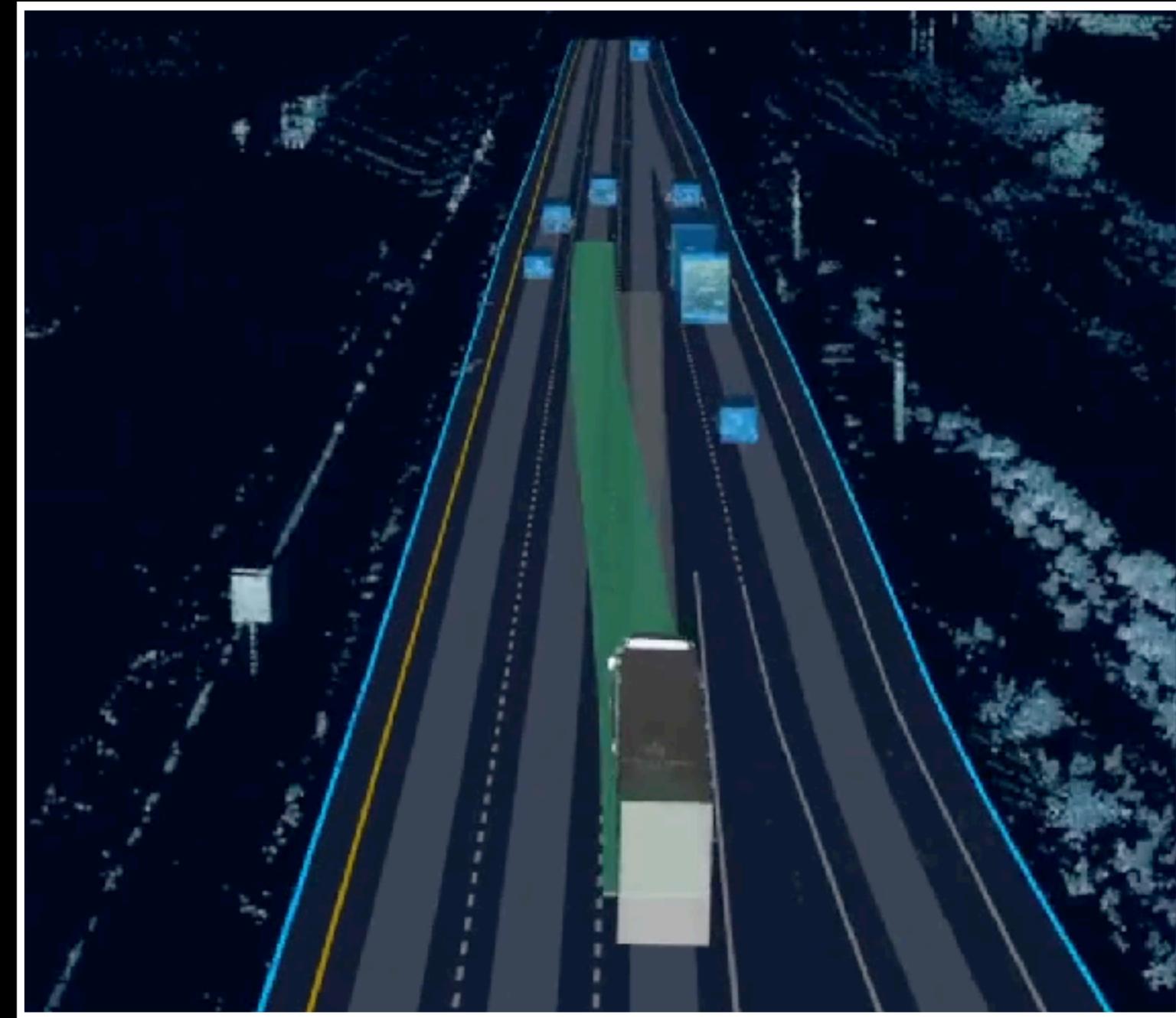
1. Collect data from a human demonstrator

$$[(s_1, a_1^*), (s_2, a_2^*), (s_3, a_3^*), \dots]$$

2. Train a policy $\pi : s_t \rightarrow a_t$ on the data

3. Check validation error on held out dataset **Why?**

Let's apply Behavior Cloning!



1. Collect data from a human demonstrator

$$[(s_1, a_1^*), (s_2, a_2^*), (s_3, a_3^*), \dots]$$

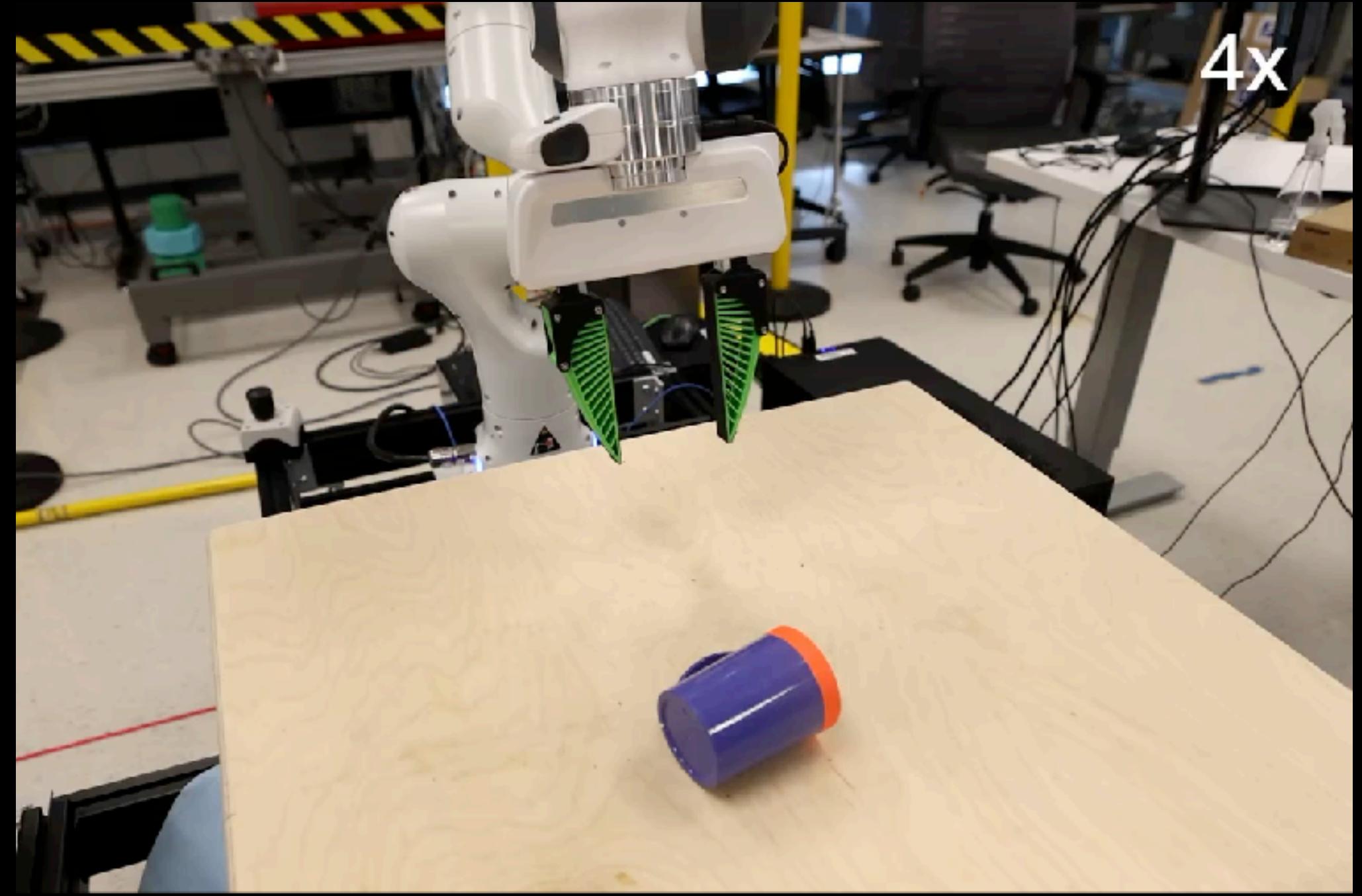
2. Train a policy $\pi : s_t \rightarrow a_t$ on the data

How do I collect demonstrations?

3. Check validation error on held out dataset

What is my state? Action? Loss?

Let's apply Behavior Cloning!



1. Collect data from a human demonstrator

$$[(s_1, a_1^*), (s_2, a_2^*), (s_3, a_3^*), \dots]$$

How do I collect demonstrations?

2. Train a policy $\pi : s_t \rightarrow a_t$ on the data

What is my state? Action? Loss?

3. Check validation error on held out dataset

Why we love Behavior Cloning

It's EASY!

If you can drive down validation error perfectly to 0,
it is *guaranteed* to do what the expert does

It may work often in practice, but ...

What do you see as practical challenges with BC?

When poll is active respond at PollEv.com/sc2582



How things go wrong with BC

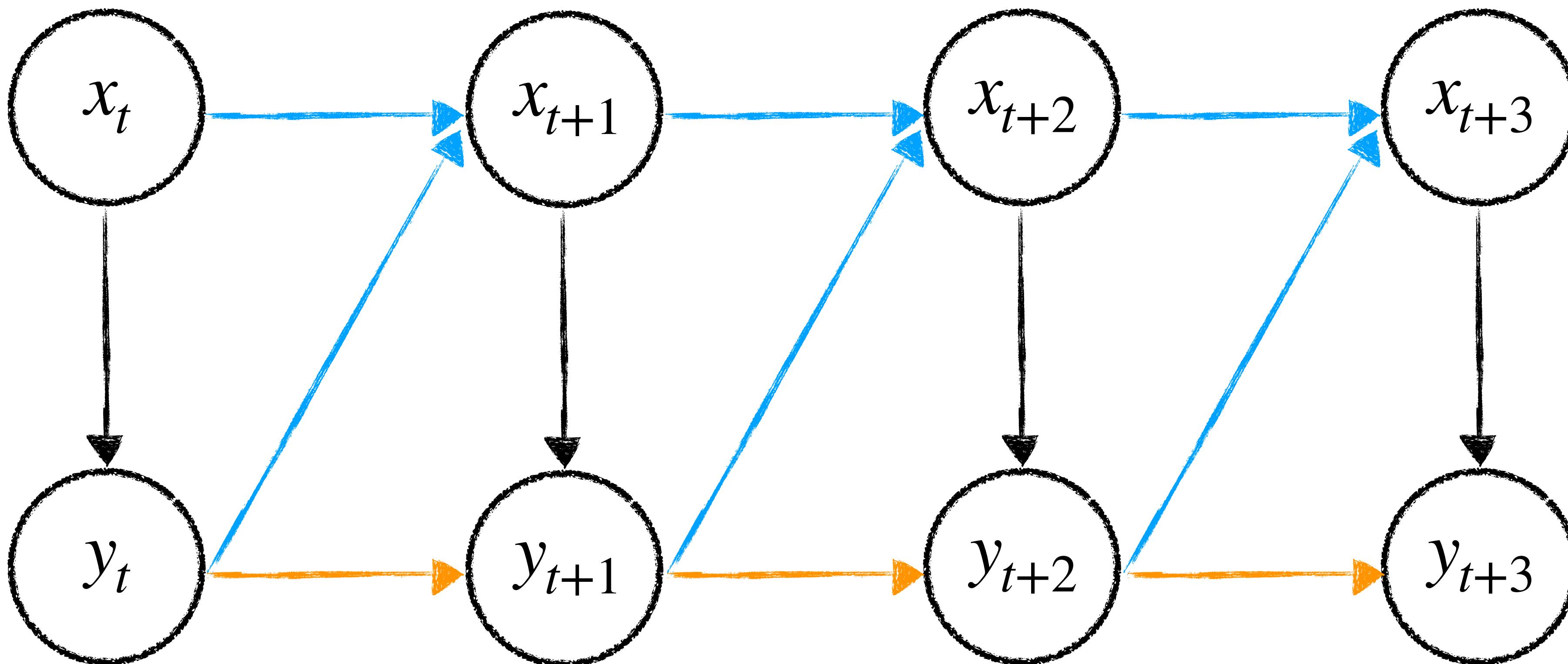






Feedback drives
covariate shift

Feedback Drives Covariate Shift



Supervised Learning assumes all datapoints are i.i.d

An old problem

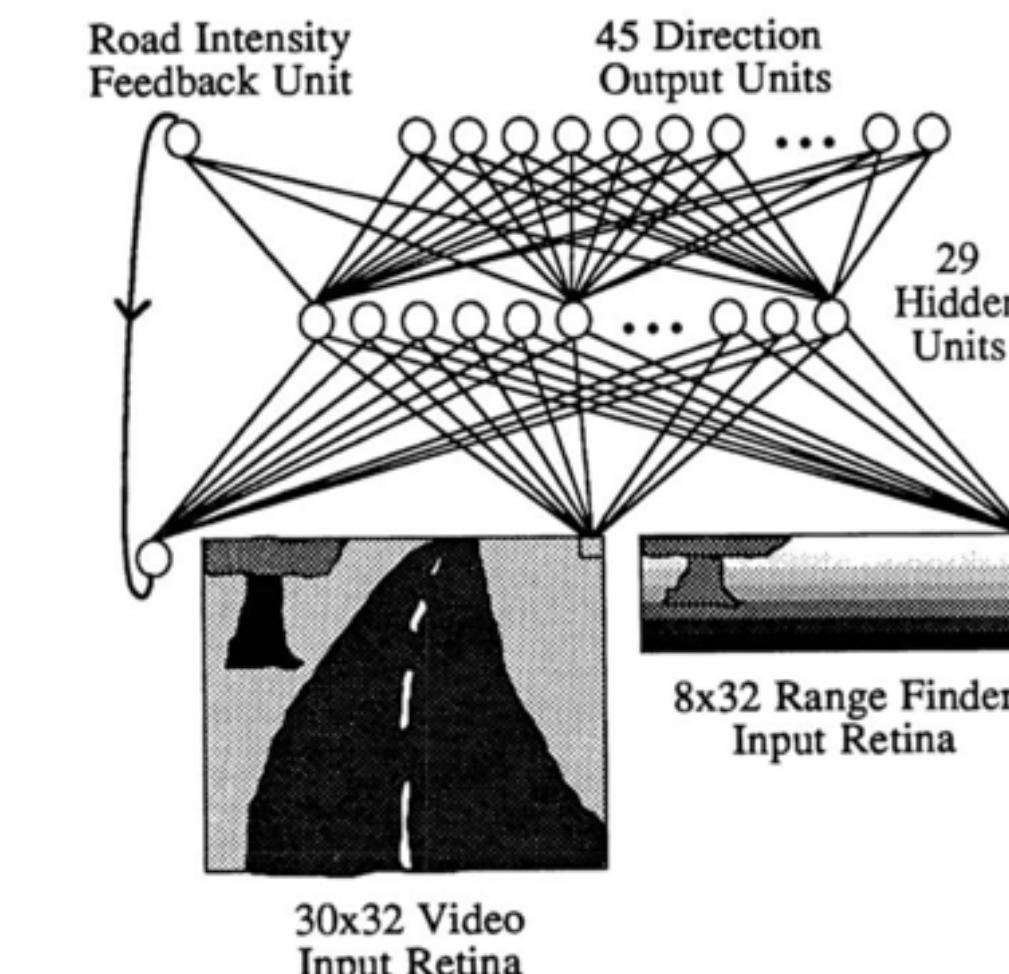


Figure 1: ALVINN Architecture

“...the network must not solely be shown examples of accurate driving, **but also how to recover** (i.e. return to the road center) once a mistake has been made.”

D. Pomerleau

ALVINN: An Autonomous Land Vehicle In A Neural Network, NeurIPS'89

Also observed by [LeCun'05]

Feedback is a pervasive problem in self-driving

“... the inertia problem. When the ego vehicle is stopped (e.g., at a red traffic light), the probability it stays static is indeed overwhelming in the training data. This creates a spurious correlation between low speed and no acceleration, inducing excessive stopping and difficult restarting in the imitative policy ...”

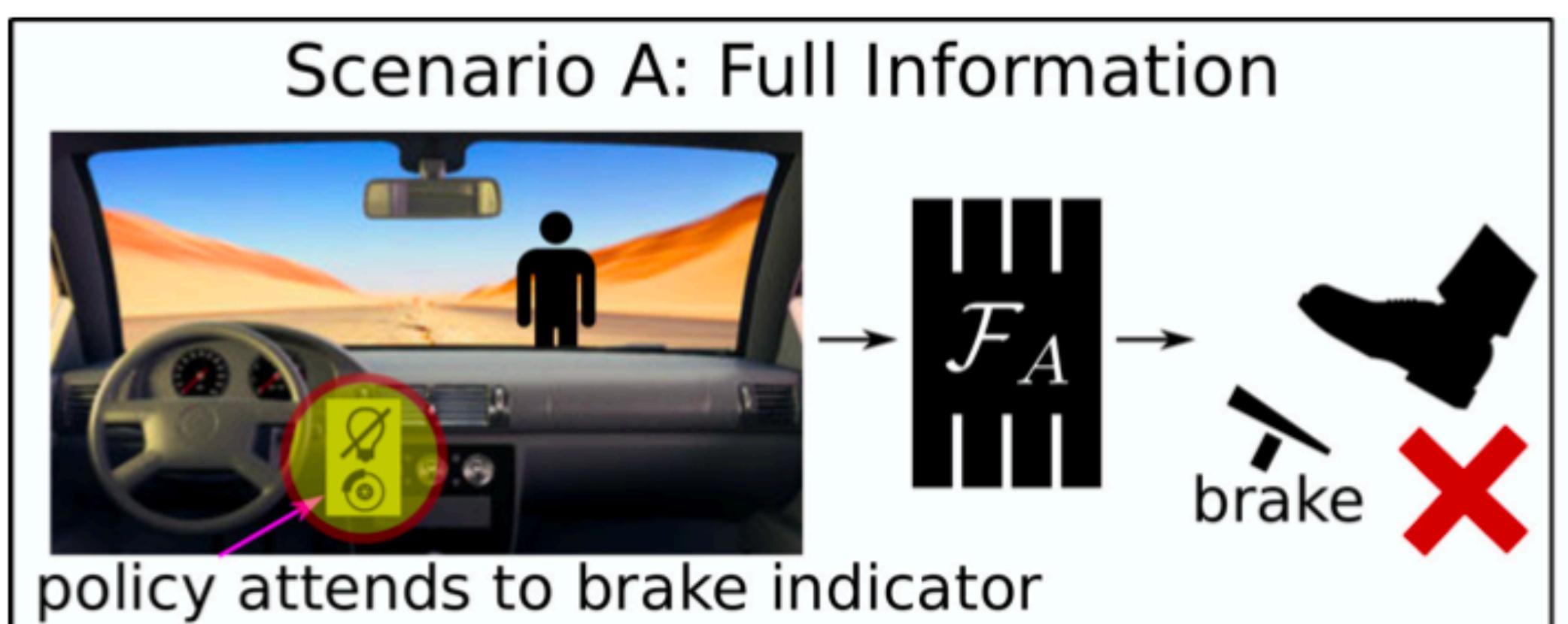
“Exploring the Limitations of Behavior Cloning for Autonomous Driving.”
F. Codevilla, E. Santana, A. M. Lopez, A. Gaidon. ICCV 2019

“... small errors in action predictions to compound over time, eventually leading to states that human drivers infrequently visit and are not adequately covered by the training data. Poorer predictions can cause a feedback cycle known as cascading errors ...”

“Imitating Driver Behavior with Generative Adversarial Networks”.
A. Kuefler, J. Morton, T. Wheeler, M. Kochenderfer, IV 2017

“... During closed-loop inference, this breaks down because the past history is from the net’s own past predictions. For example, such a trained net may learn to only stop for a stop sign if it sees a deceleration in the past history, and will therefore never stop for a stop sign during closed-loop inference ...”

“ChauffeurNet: Learning to Drive by Imitating the Best and Synthesizing the Worst”. M. Bansal, A. Krizhevsky, A. Ogale, Waymo 2018



“Causal Confusion in Imitation Learning”.
P. de Haan, D. Jayaraman, S. Levine, NeurIPS ‘19



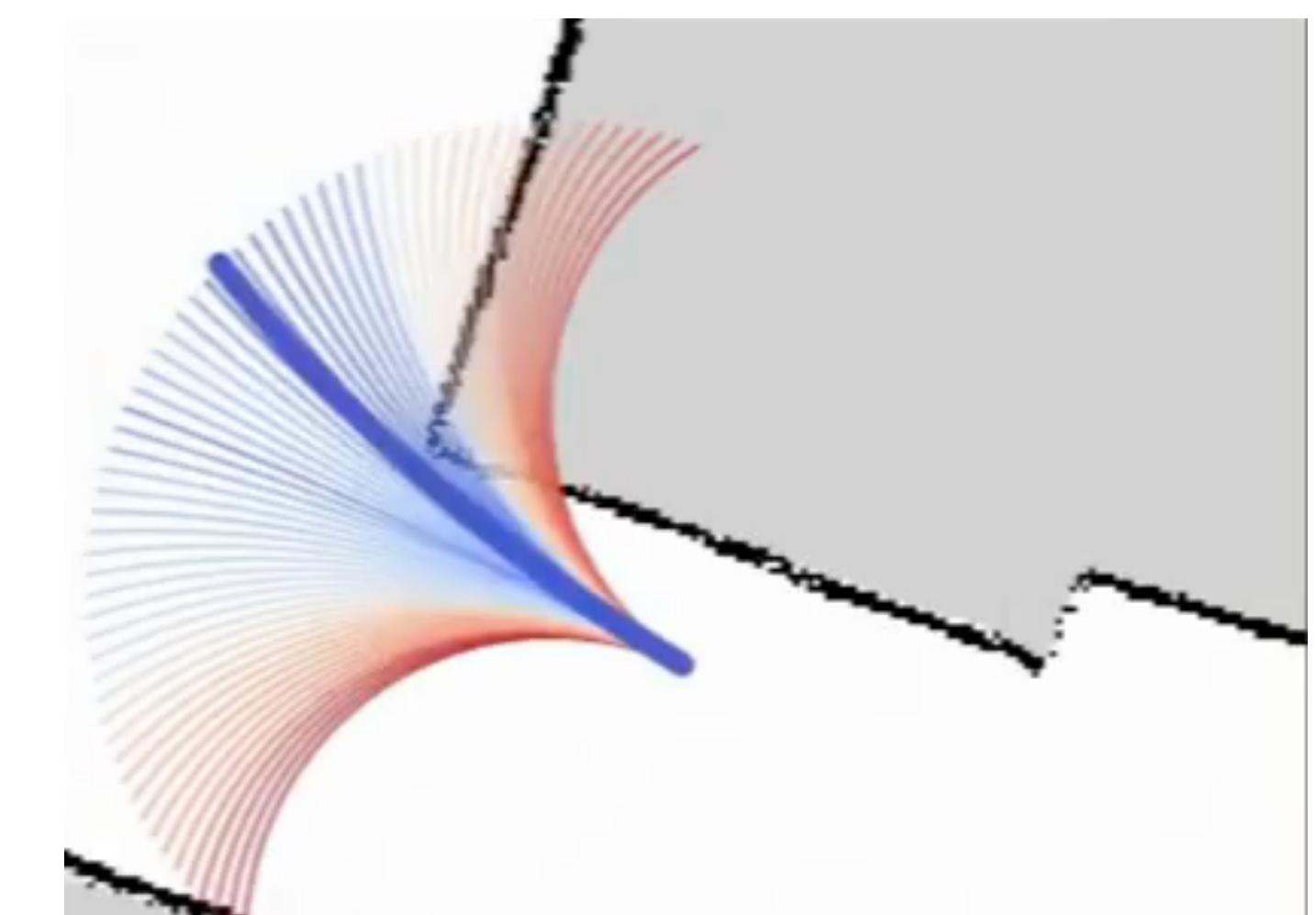
Feedback is an old adversary!



[SCB+ RSS'20]



Demonstration

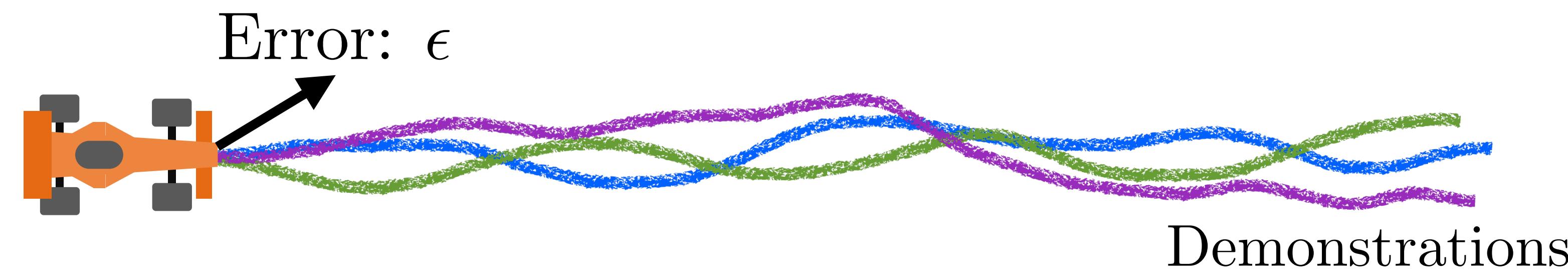


Learnt policy

Behavior Cloning crashes into a wall

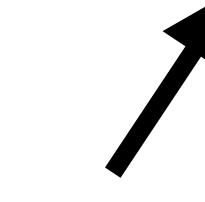


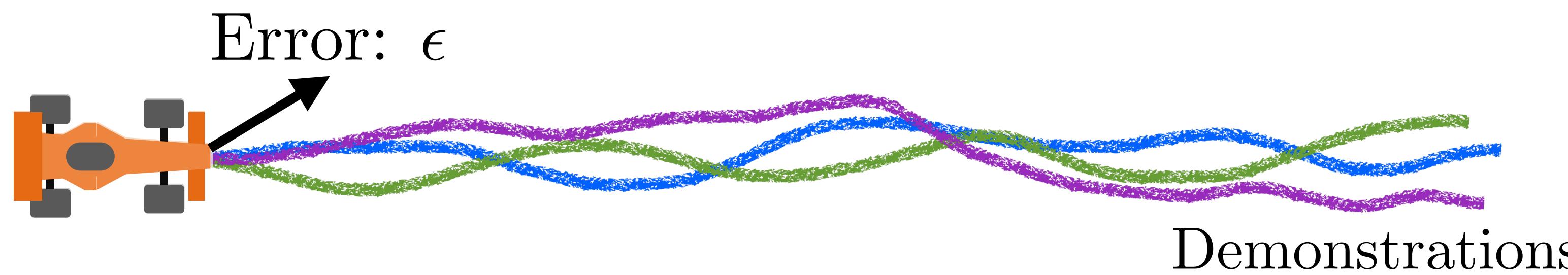
Why did the robot crash?



Why did the robot crash?



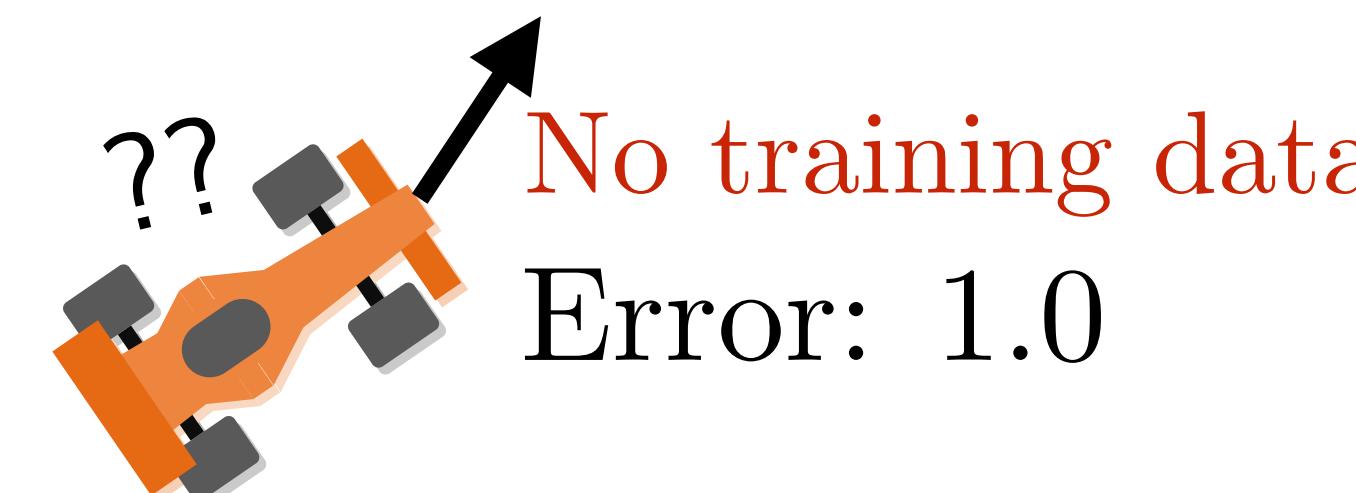
??  No training data
Error: 1.0



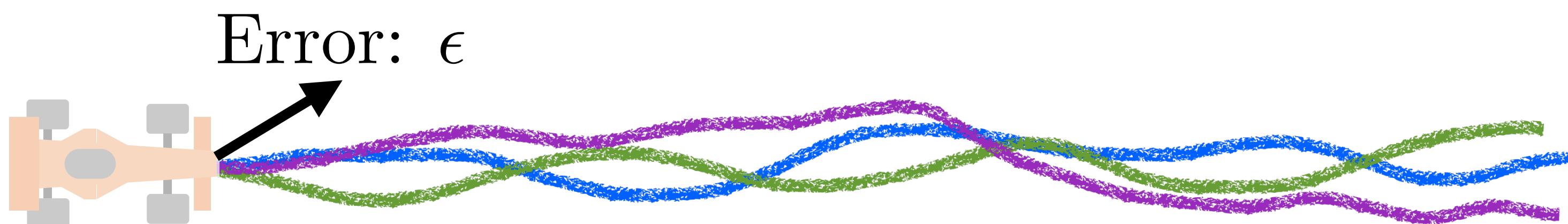
Why did the robot crash?



No training data
Error: 1.0



No training data
Error: 1.0



Demonstrations

Can we mathematically
quantify how much
worse BC is compared
to the demonstrator?



First, let's define **performance** of a policy

$$J(\pi) = \mathbb{E}_{\substack{a_t \sim \pi(s_t) \\ s_{t+1} \sim \mathcal{T}(s_t, a_t)}} \left[\sum_{t=0}^{T-1} c(s_t, a_t) \right]$$

(Performance)

Second, let's define performance difference

$$J(\pi) - J(\pi^*)$$

(Performance
of my learner) (Performance
of my demonstrator)

We want to *minimize* the performance difference

How low can we drive performance difference?

Let's say my learner is not perfect and can only drive down training / validation error to be ϵ



The **best** we can hope for is that error grows **linearly** in time

$$J(\pi) - J(\pi^*) \leq O(\epsilon T)$$



The **worst** case is if error **compounds quadratically** in time

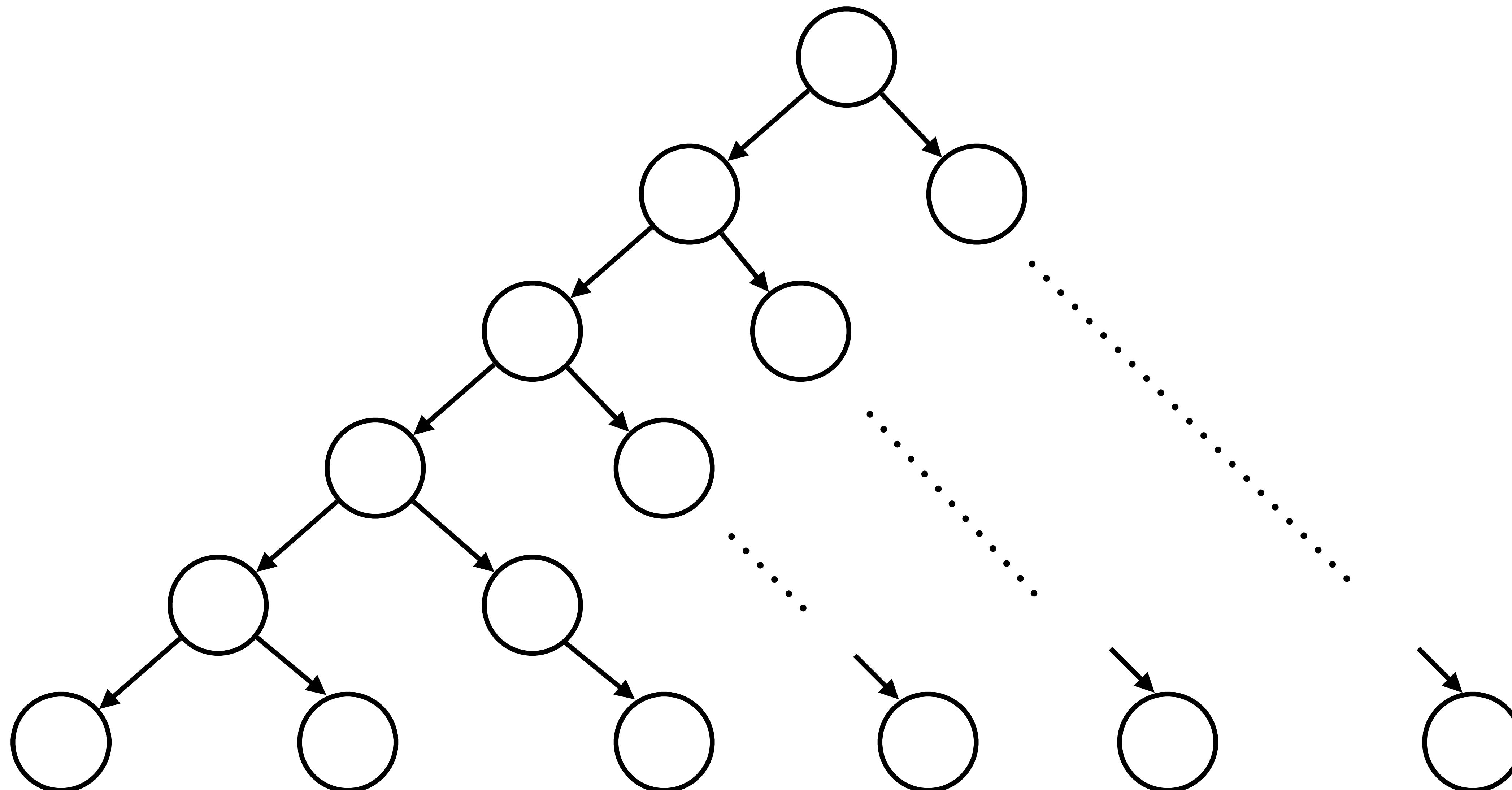
$$J(\pi) - J(\pi^*) \leq O(\epsilon T^2)$$

Behavior cloning hits the worst case!

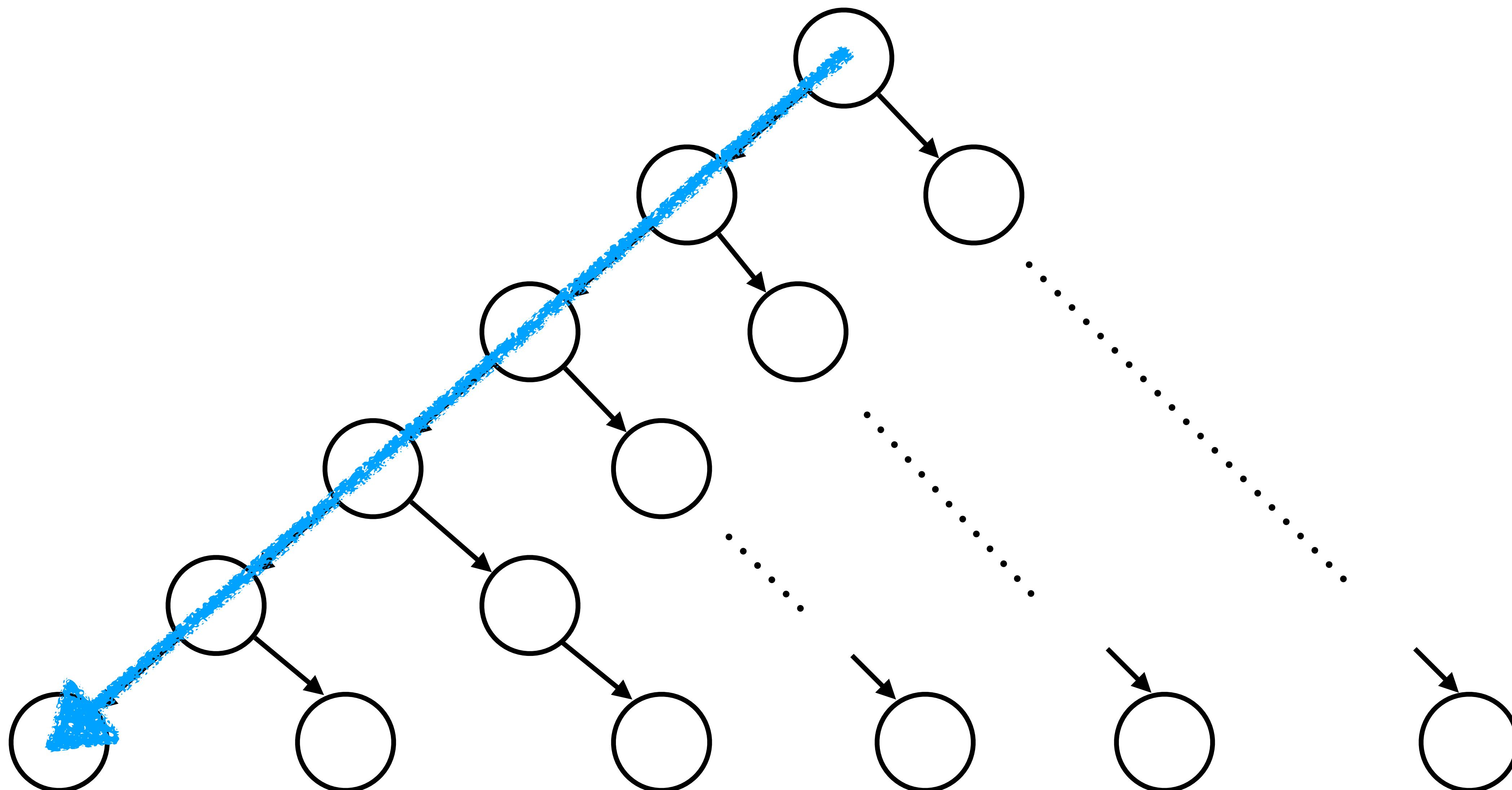
*There exists an MDP where BC
has a performance difference of $O(\epsilon T^2)$*

We are going to such a MDP right now,
and you will see more in A1!

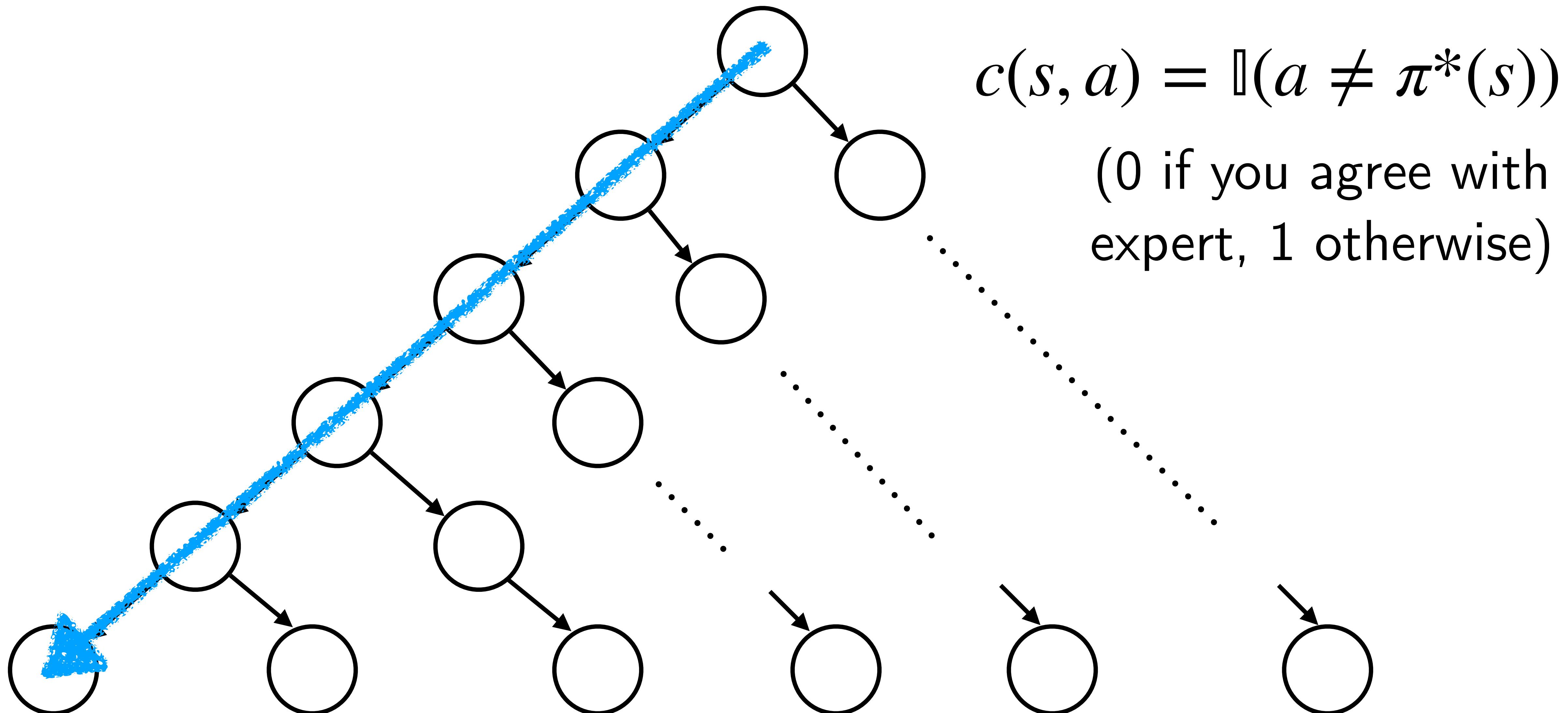
A Tree MDP



The demonstrator always takes a left

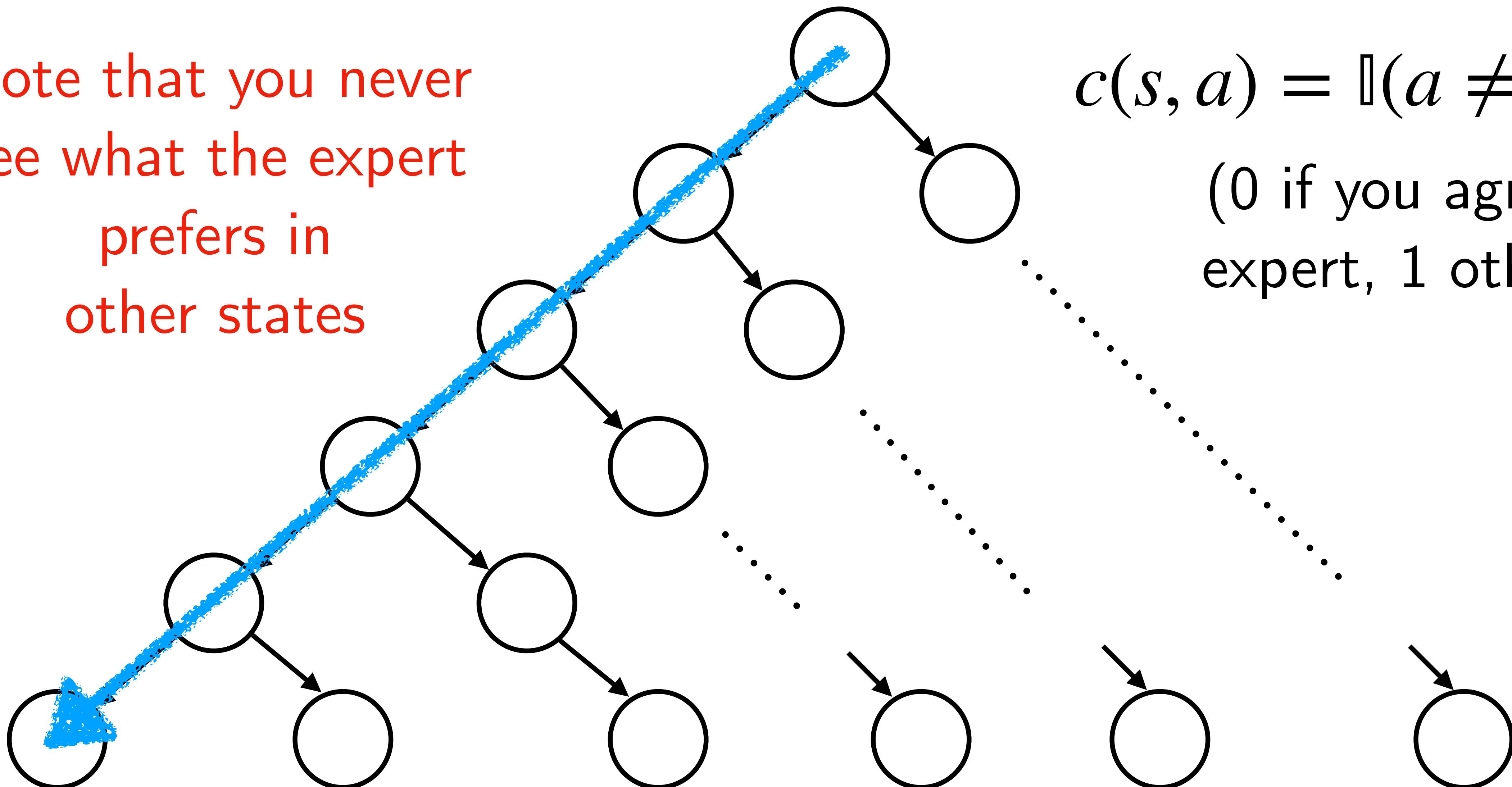


Assume the following cost function



Assume the following cost function

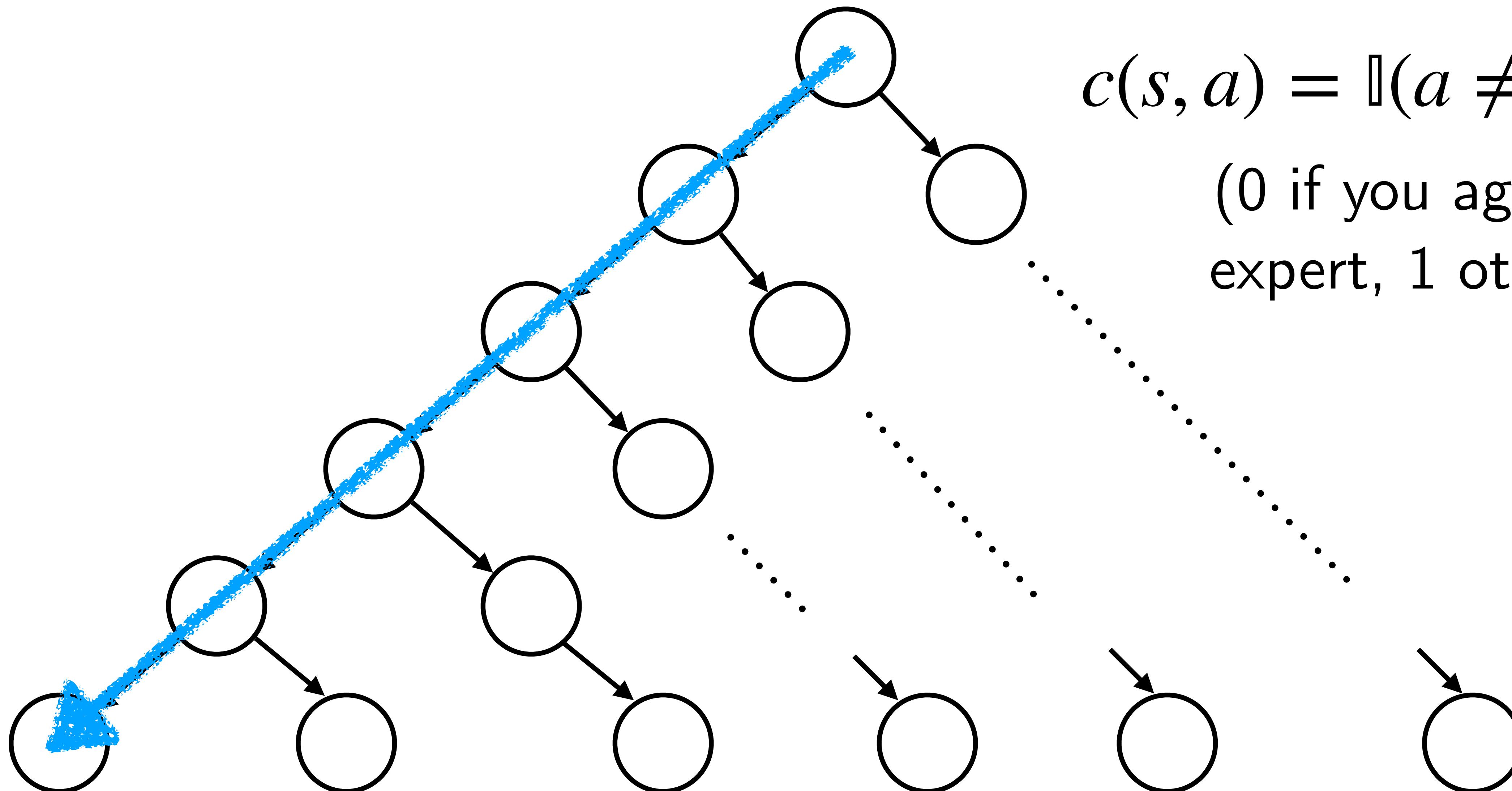
Note that you never
see what the expert
prefers in
other states



$$c(s, a) = \mathbb{I}(a \neq \pi^*(s))$$

(0 if you agree with
expert, 1 otherwise)

Show that BC has a performance difference of $O(\epsilon T^2)$



$$c(s, a) = \mathbb{I}(a \neq \pi^*(s))$$

(0 if you agree with expert, 1 otherwise)