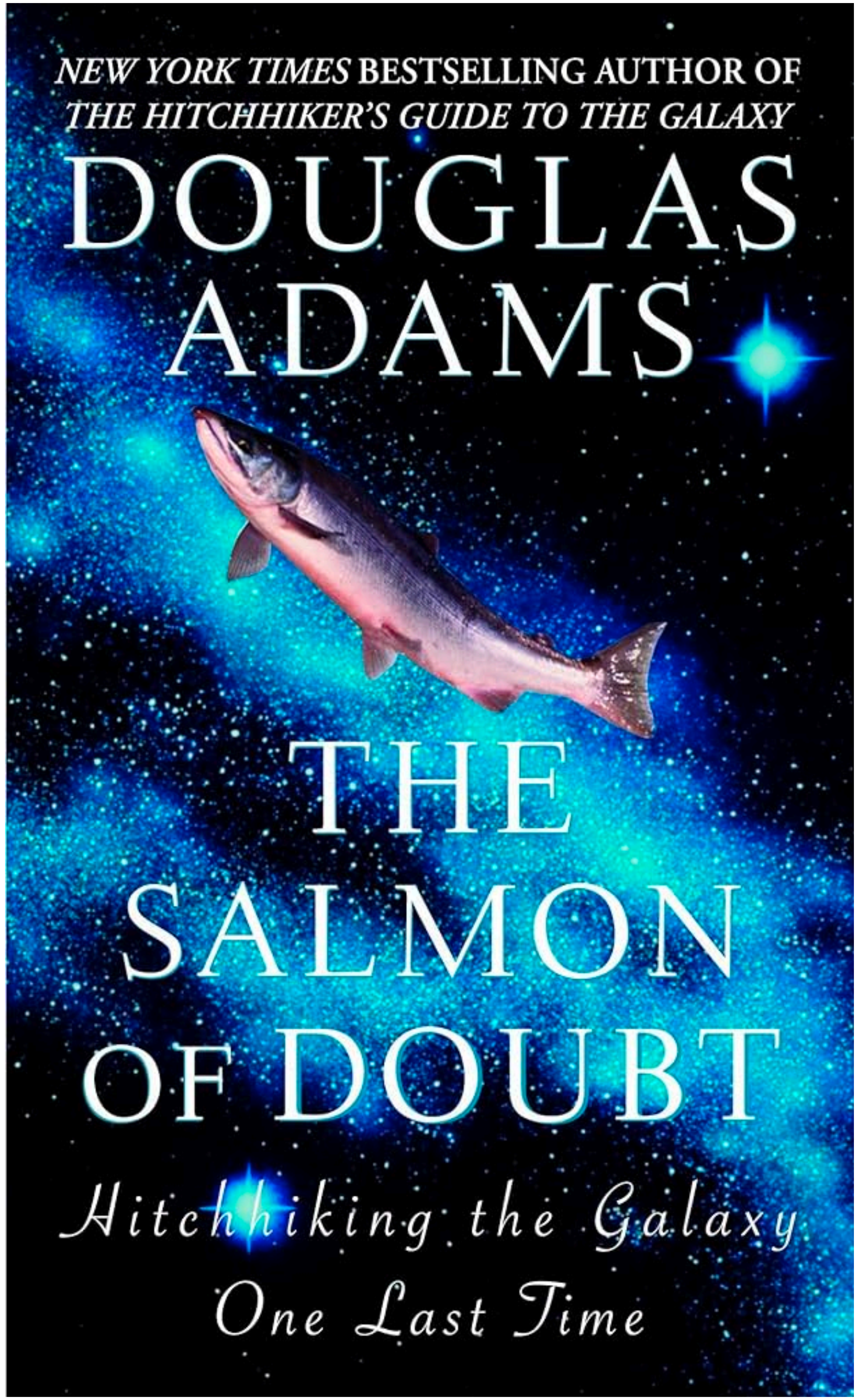
Multi-Agent Forecasting and Imitation Learning

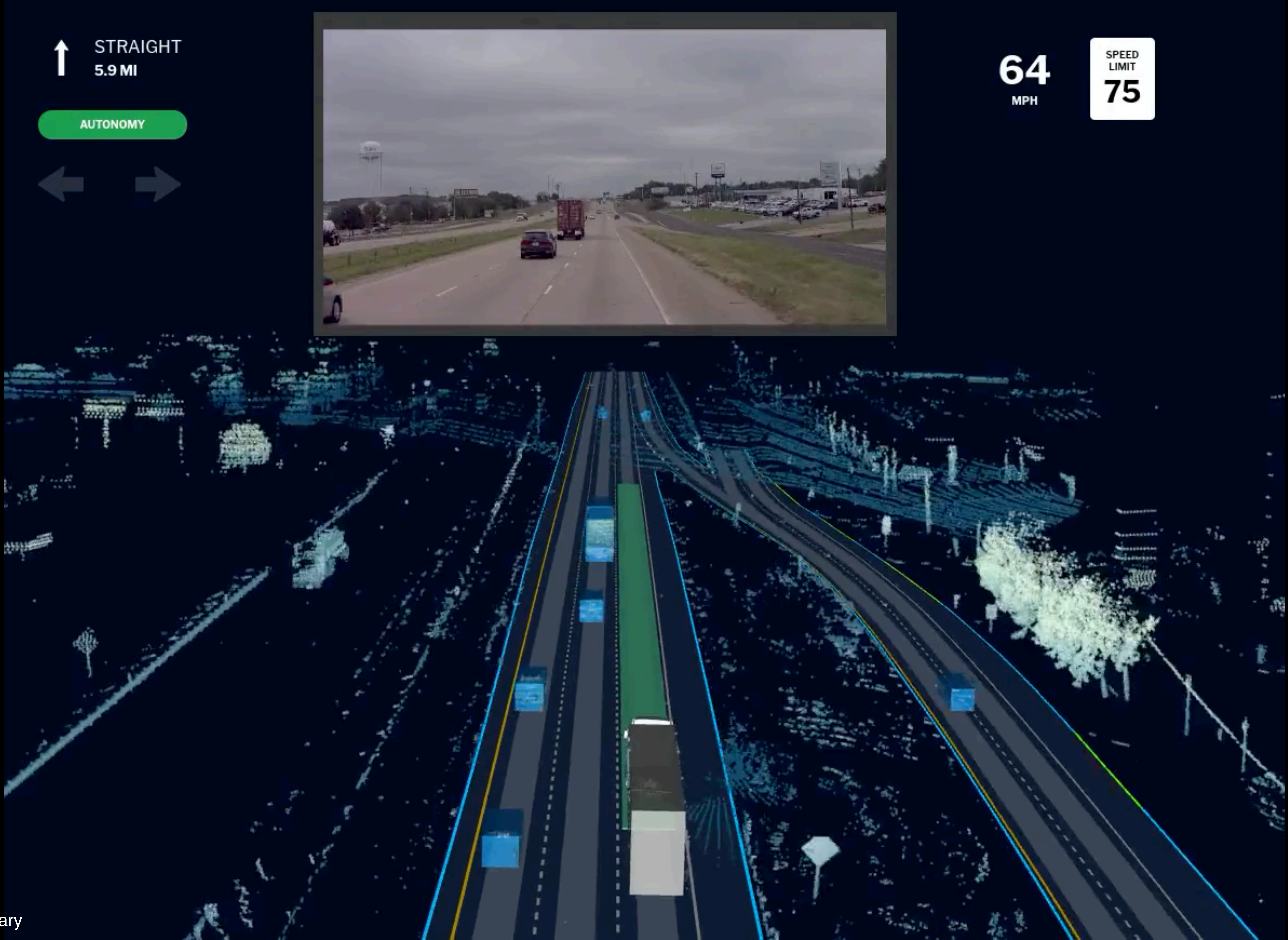
Sanjiban Choudhury

Cornell Bowers CIS
Computer Science

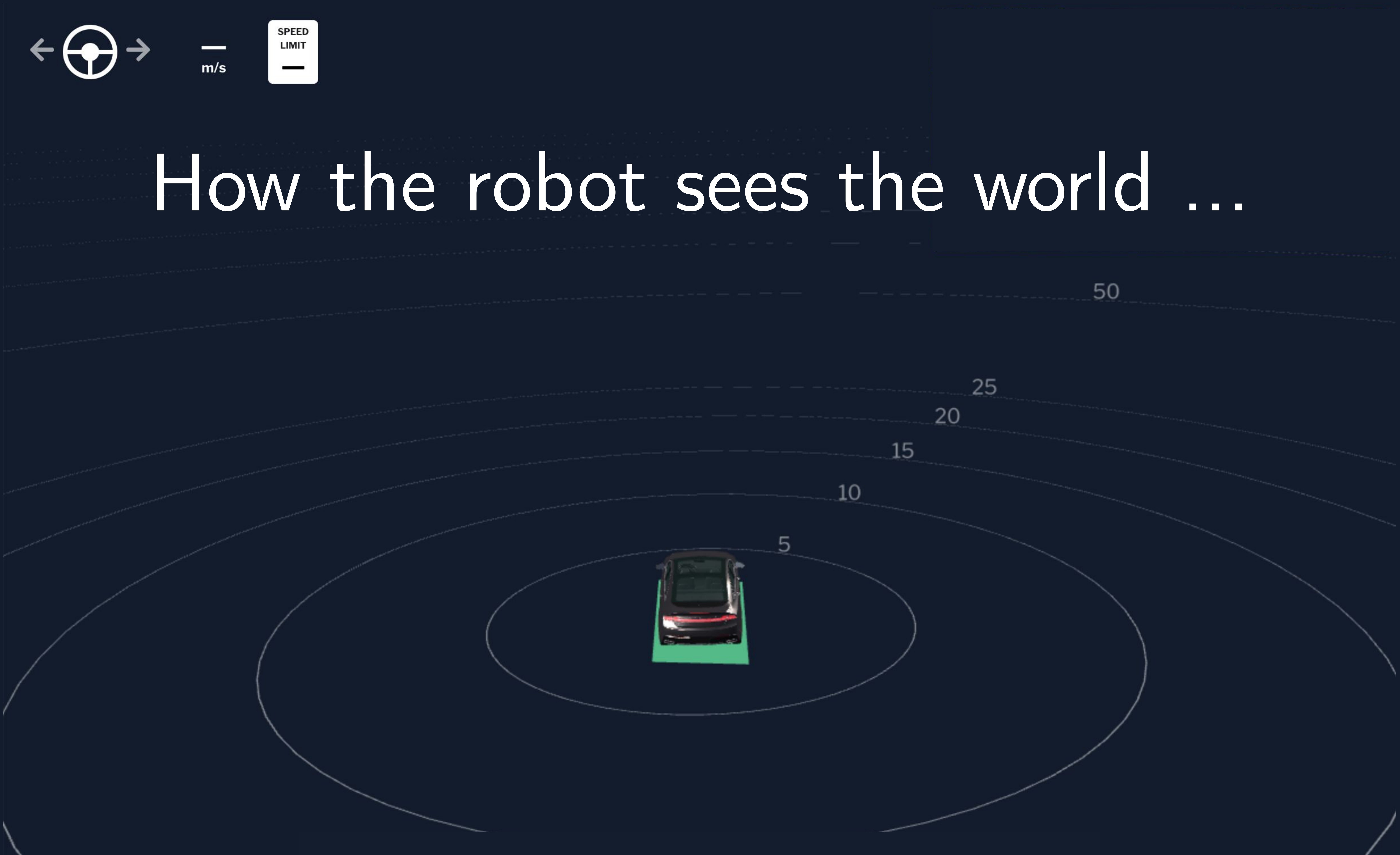
"Trying to predict the future is a mug's game..."




30.0 m
20.0 m
10.0 m
1541016641.475 20181031.175444Z.mkz-07.mkz.V0-592_MDC


STRAIGHT
5.9 MI
AUTONOMY
64
MPH
SPEED LIMIT
75

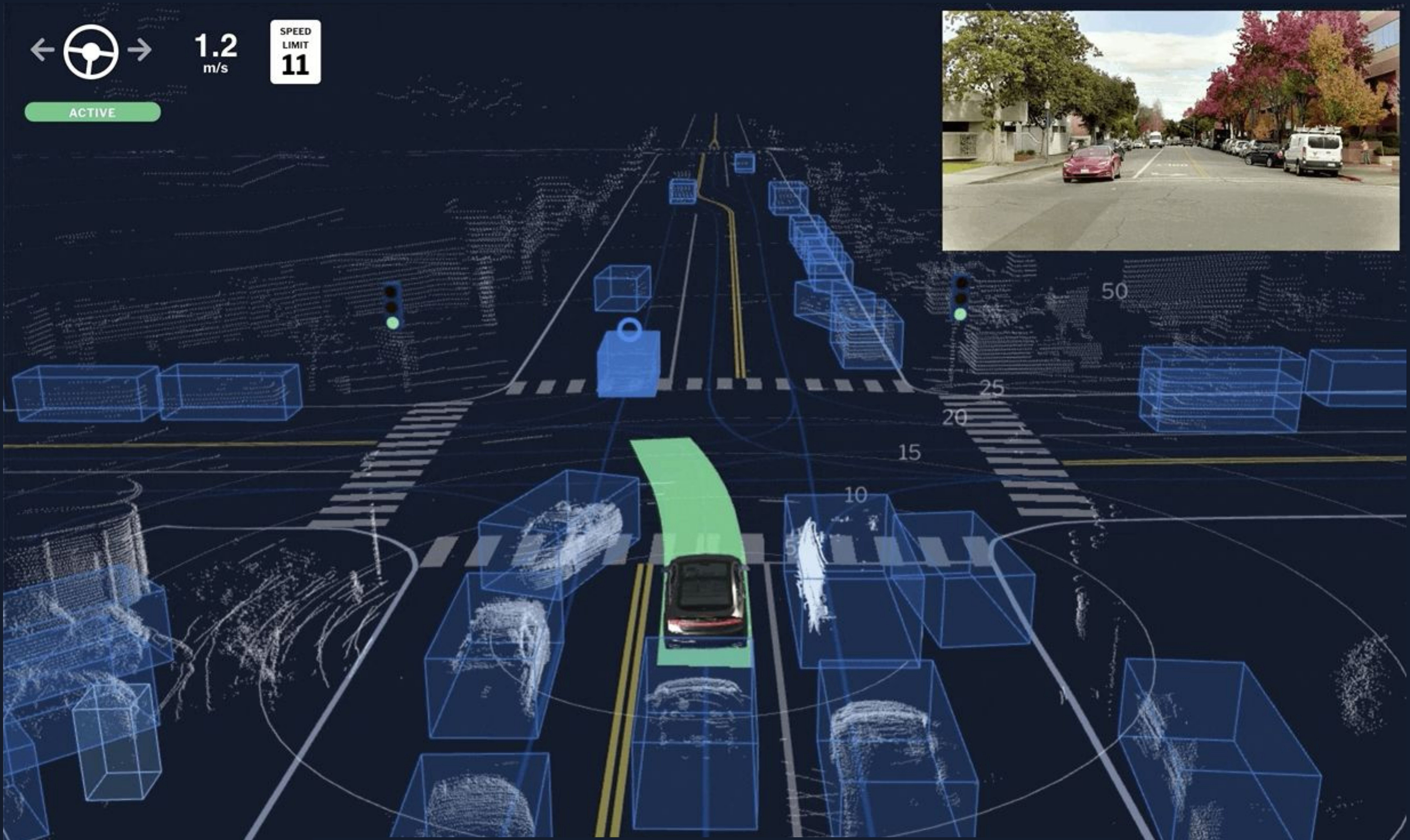
How the robot sees the world ...



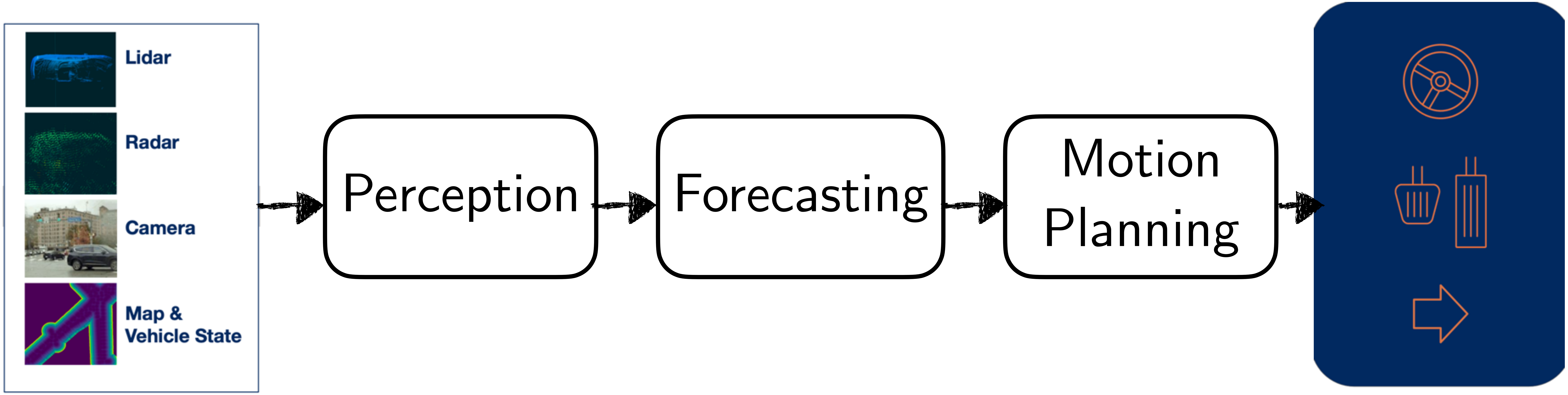

0.0
m/s
SPEED
LIMIT
11
ACTIVE
50
25
20
15
10

0.0
m/s
SPEED
LIMIT
11
ACTIVE
50
25
20
15
10
5


0.0
m/s
SPEED
LIMIT
11
ACTIVE
50
25
20
15
10


1.2
m/s
SPEED
LIMIT
11
ACTIVE
50
25
20
15
10

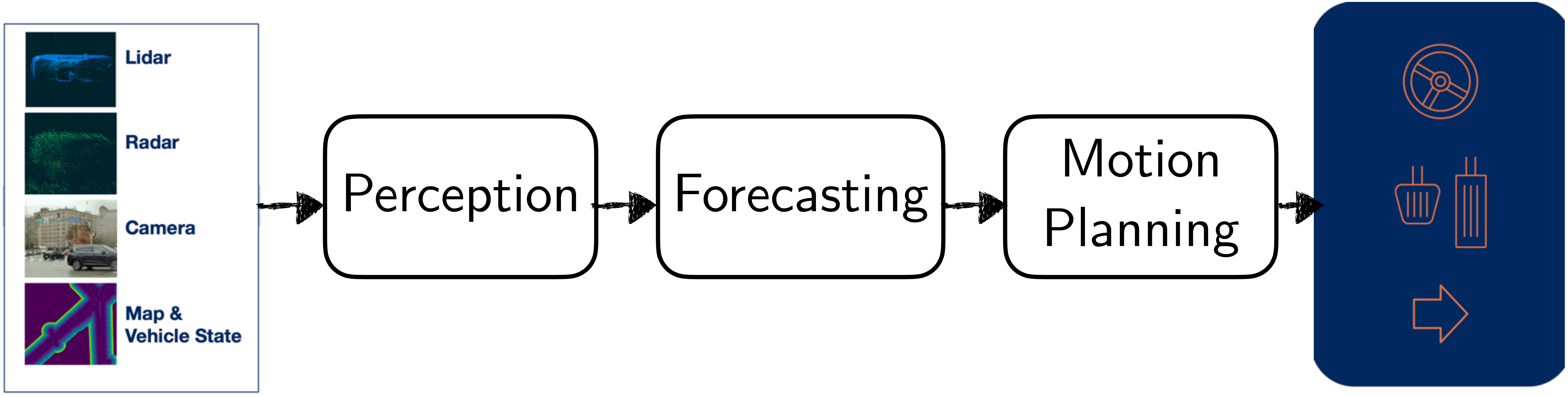
Traditional Architecture



Raw sensor data

Control actions

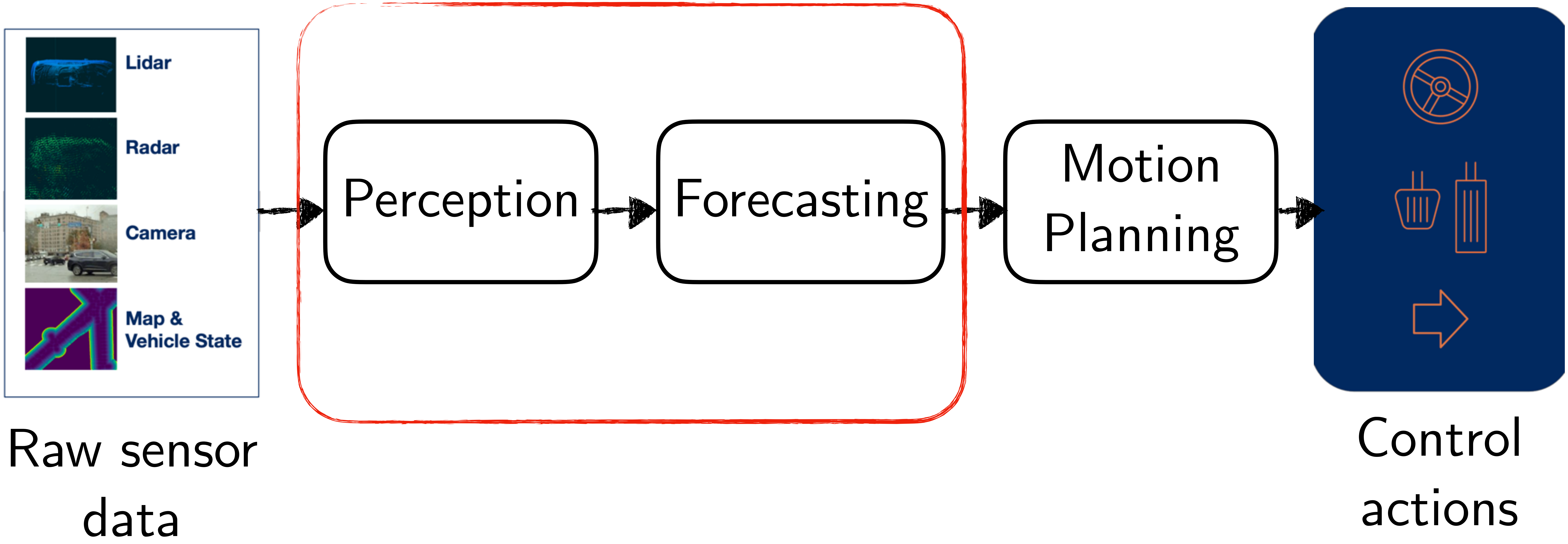
Is having cascaded blocks a good idea?



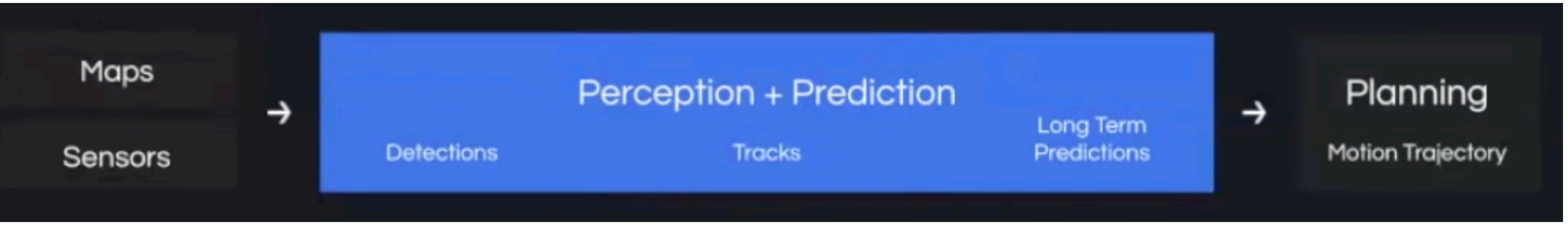
Raw sensor data

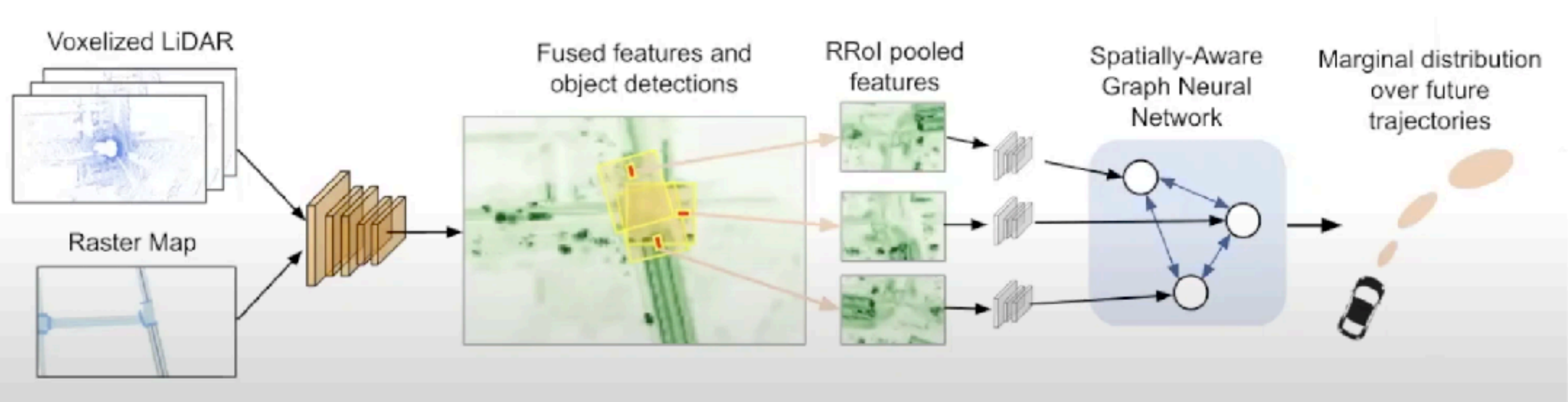
Control actions

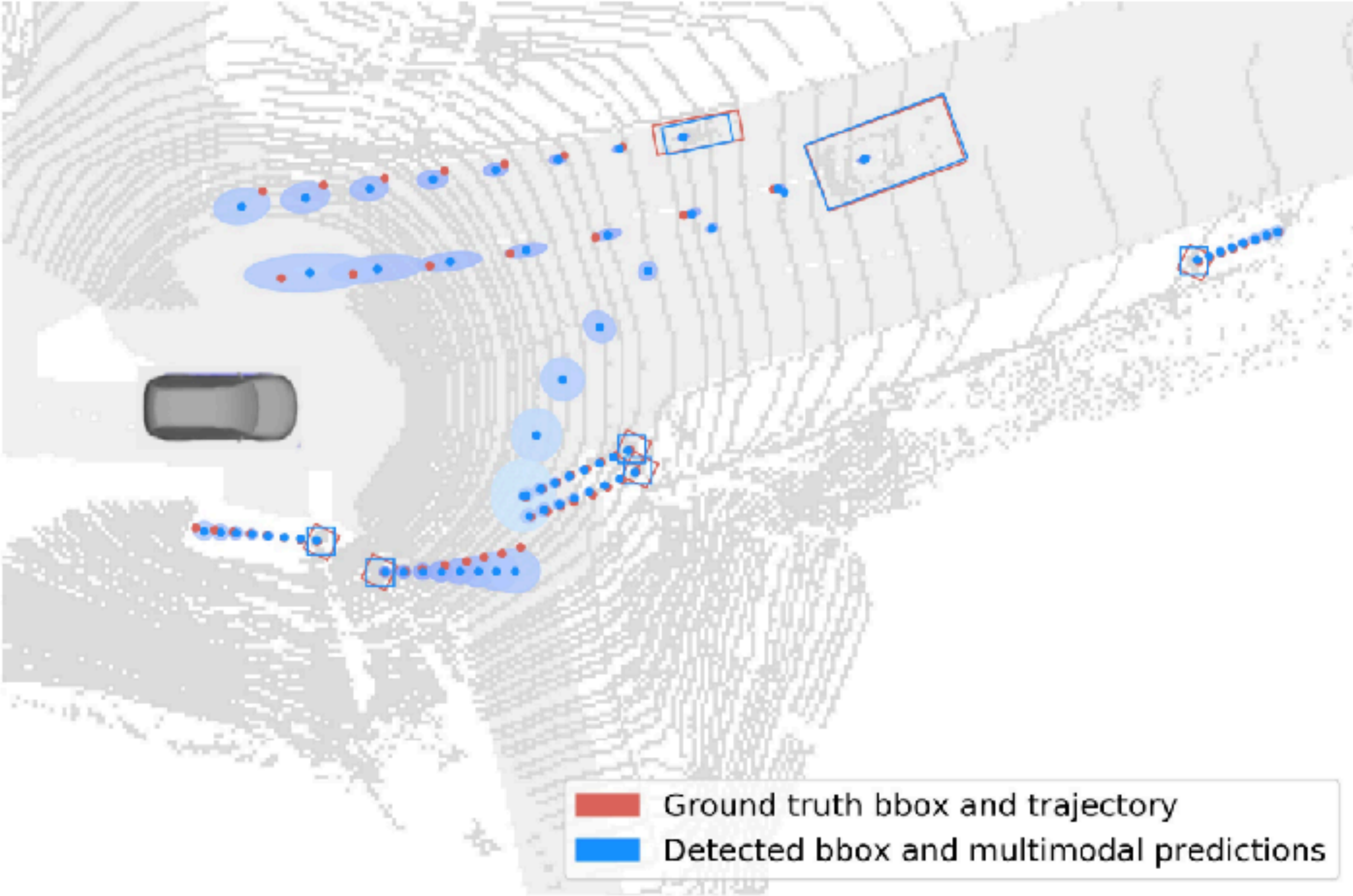
Lots of work on perception+forecasting



Lots of work on perception+forecasting







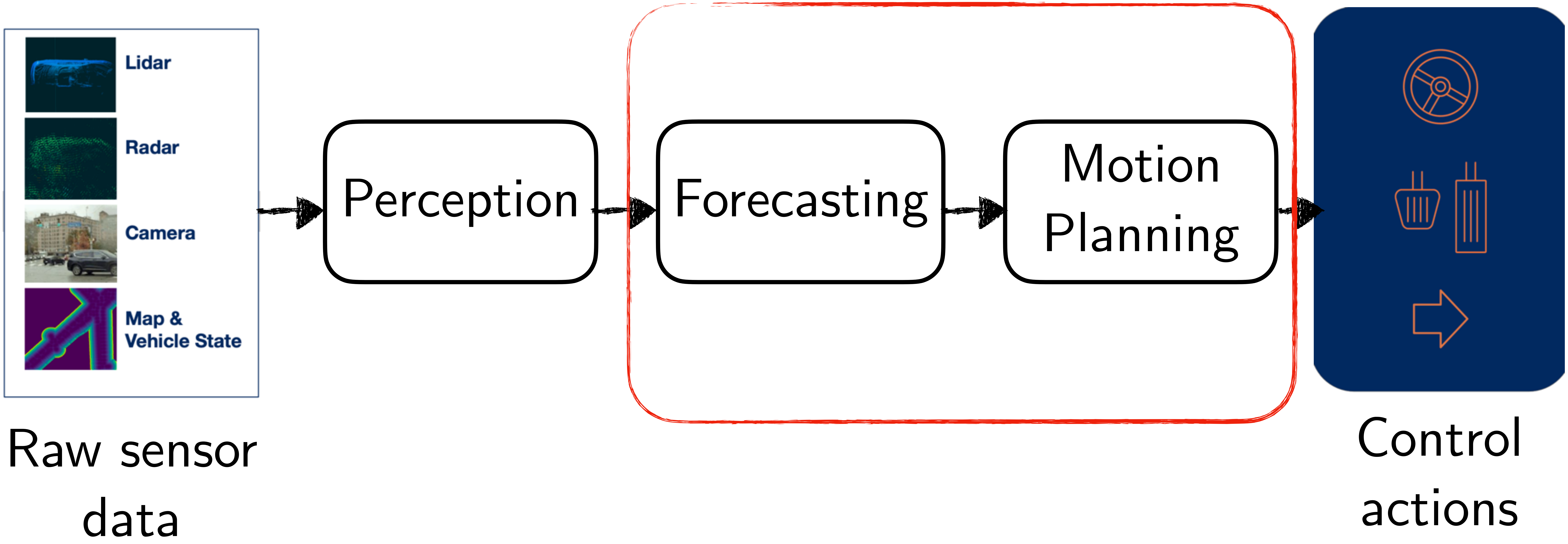
SPAGNN: Spatially-Aware Graph Neural Networks for Relational Behavior Forecasting from Sensor Data

Sergio Casas[1,2], Cole Gulino[1], Renjie Liao[1,2], Raquel Urtasun[1,2]
Uber Advanced Technologies Group[1], University of Toronto[2]
{sergio.casas, cgulino, rjliao, urtasun}@uber.com

MultiXNet: Multiclass Multistage Multimodal Motion Prediction

Nemanja Djuric, Henggang Cui, Zhaoen Su, Shangxuan Wu, Huahua Wang, Fang-Chieh Chou, Luisa San Martin, Song Feng, Rui Hu, Yang Xu, Alyssa Dayan, Sidney Zhang, Brian C. Becker, Gregory P. Meyer, Carlos Vallespi-Gonzalez, Carl K. Wellington
Uber Advanced Technologies Group
{ndjuric, hcui2, suzhaoen, shangxuan.wu, anteaglewang, fchou, luisasm}@uber.com
{songf, rui.hu, yang.xu, ada, sidney, bbecker, gmeyer, cvallespi, cwellington}@uber.com

What about forecasting+planning?



Forecasting is a very active area of research!

types.

Leaderboard

This leaderboard only displays submissions made on or after March 15, 2023, when the 2023 Waymo Open Dataset Challenges start.

With the latest v1.2.0 of the Motion Dataset, Lidar data is now available for the 1s history data.

To view a ranked leaderboard for the 2023 Motion Prediction Challenge, with Soft mAP as the ranking metric, please select "Show results with Soft mAP only" under the "Soft mAP" column. To view a ranked leaderboard with mAP as the ranking metric, please select "Show all" under the "Soft mAP" column, and click on the "mAP" column to sort. Submissions before March 9, 2022 will not have a Soft mAP score.

Note: the rankings displayed on this leaderboard may not accurately reflect the final rankings for this Challenge.

Method Name	Lidar data for training	Object Type (All)	Evaluation Time (Avg)	Soft mAP (Show rest)	mAP	minADE	minFDE	Miss Rate	Overlap Rate	Date (Pacific Daylight Time)
MGTR_ens		All	Avg	0.4764	0.4658	0.5825	1.2009	0.1258	0.1270	2023-09-15 19:06
MTR++_Ens		All	Avg	0.4738	0.4634	0.5581	1.1166	0.1122	0.1276	2023-05-23 15:37
MGTR		All	Avg	0.4599	0.4505	0.5918	1.2135	0.1298	0.1275	2023-09-14 21:18
GTR_ens		All	Avg	0.4518	0.4428	0.5855	1.2056	0.1296	0.1277	2023-05-25 02:58
EDA_single		All	Avg	0.4510	0.4401	0.5718	1.1702	0.1169	0.1266	2023-08-07 07:23
IAIR+		All	Avg	0.4480	0.4347	0.5783	1.1679	0.1238	0.1263	2023-05-23 23:56
MTR++		All	Avg	0.4414	0.4329	0.5906	1.1939	0.1298	0.1281	2023-05-23 12:31
GTR-R36		All	Avg	0.4384	0.4255	0.6005	1.2225	0.1330	0.1279	2023-05-23 20:39
GTR		All	Avg	0.4365	0.4230	0.5871	1.2096	0.1309	0.1272	2023-05-16 17:50
DM		All	Avg	0.4362	0.4301	0.6293	1.2723	0.1473	0.1364	2023-05-23 23:39
DMotion		All	Avg	0.4361	0.4240	0.6092	1.2247	0.1332	0.1264	2023-10-17 23:19
MPTr+		All	Avg	0.4267	0.4130	0.5963	1.2060	0.1318	0.1265	2023-05-22 12:07
MPTr		All	Avg	0.4158	0.4018	0.6093	1.2232	0.1336	0.1279	2023-05-18 09:22
vtstats		All	Avg	0.4093	0.3976	0.6039	1.2231	0.1364	0.1289	2023-05-23 08:32

Argoverse Motion Forecasting Competition

★ 219

Organized by: argoai-argoverse
Starts on: Sep 27, 2019 8:00:00 PM EST (GMT - 5:00)
Ends on: Dec 1, 2099 6:59:59 PM EST (GMT - 5:00)

Overview Evaluation Phases Participate Leaderboard Discuss

Leaderboard

Phase: CVPR 2021 competition, Split: Test Split

Order by metric

B - Baseline * - Private V - Verified Visible Metrics

Rank	Participant team	brier-minFDE (K=6) (↑)	minFDE (K=6) (↑)	MR (K=6) (↑)	minADE (K=6) (↑)	DAC (K=6) (↑)	minFDE (K=1) (↑)	MR (K=1) (↑)	minADE (K=1) (↑)	La su at
1	SEPT-IDLab (SEPT)	1.6820	1.0566	0.1032	0.7282	0.9922	3.1777	0.5154	1.4412	3 r
2	HPNet	1.6923	1.1020	0.1110	0.7547	0.9879	3.4536	0.5256	1.5893	4 r
3	QCNet-AV1 (QCNet)	1.6934	1.0666	0.1056	0.7340	0.9887	3.3420	0.5257	1.5234	8 r
4	ProphNet (ProphNet)	1.6942	1.1337	0.1101	0.7623	0.9893	3.2628	0.5261	1.4910	1 y
5	lnkerr (Proln_ensemble)	1.7076	1.1237	0.1164	0.7762	0.9900	3.3144	0.5328	1.5113	2 r
6	hsk (test)	1.7178	1.0736	0.1064	0.7441	0.9898	3.4791	0.5447	1.5909	1 r
7	FFINet (FFINet)	1.7286	1.1213	0.1124	0.7606	0.9875	3.3611	0.5431	1.5327	1 y
8	USTB_yhf (distprob_62)	1.7305	1.1293	0.1084	0.7996	0.9903	3.3579	0.5390	1.5388	3 r

Forecasting is built
on shaky foundations

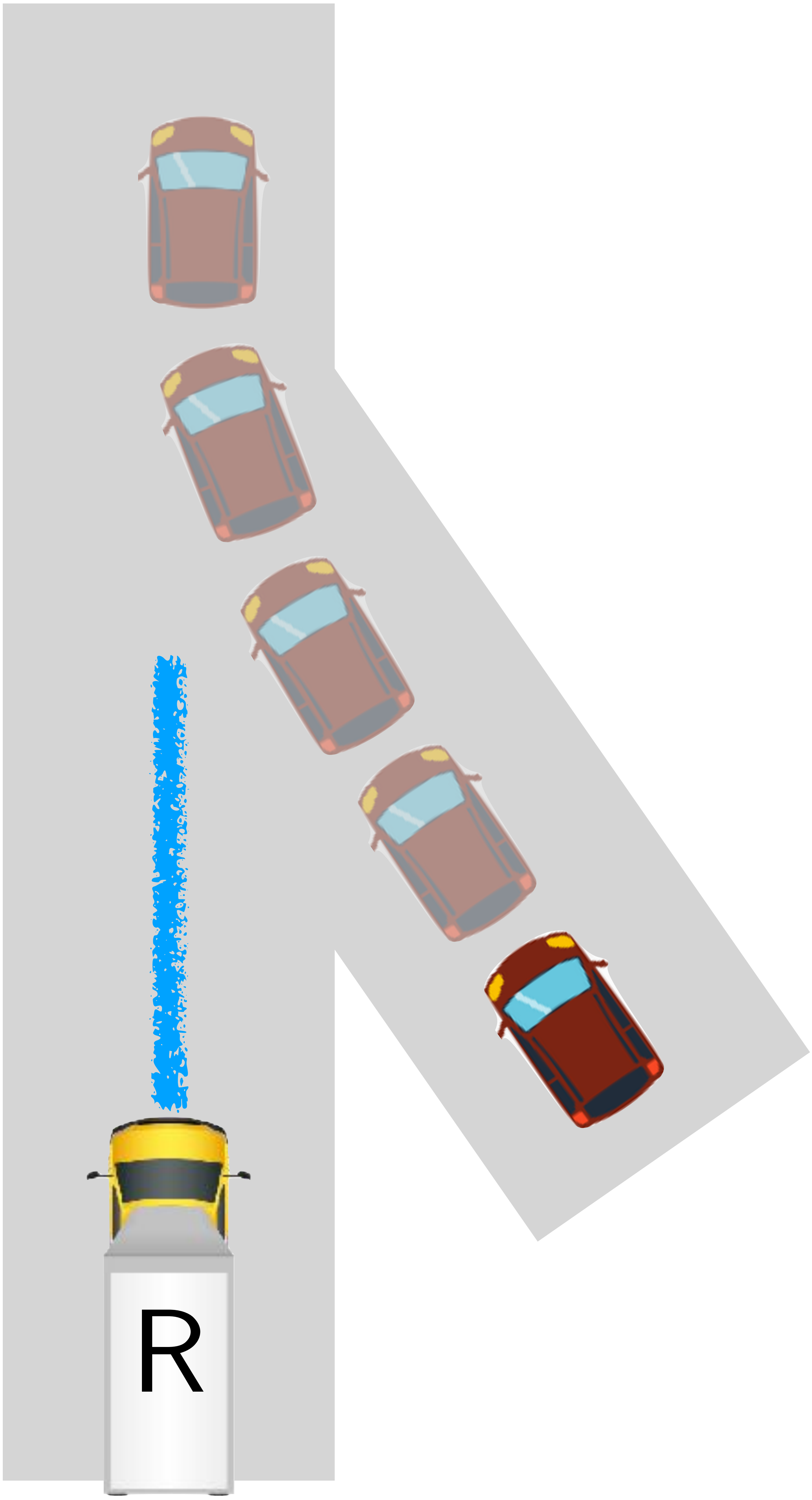
Shaky foundations of forecasting

Are we using the right model?

Are we collecting data correctly?

Are we using the right loss?

Example: Learning forecasts for merging actors



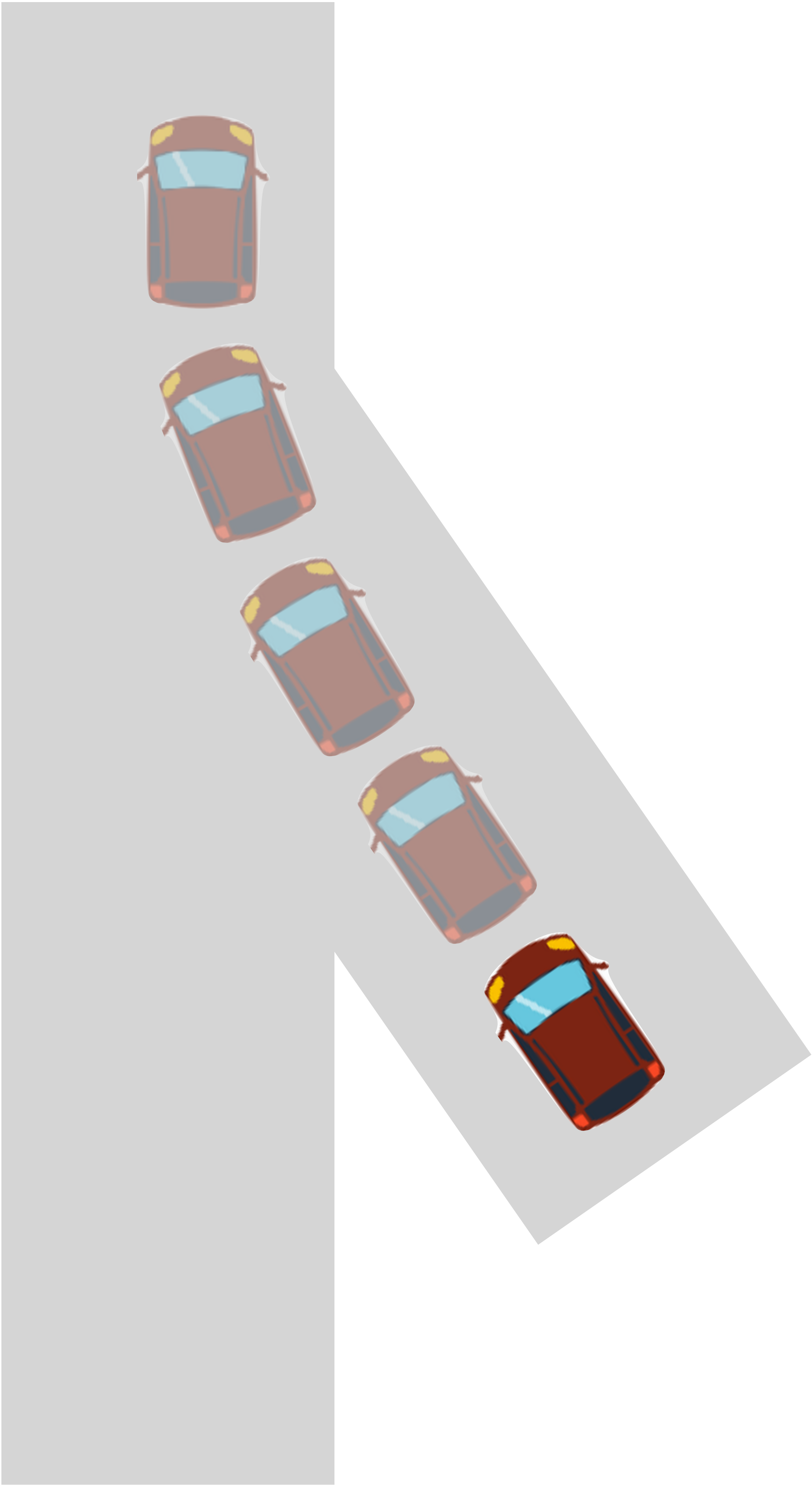
Goal

1. Predict 5s future trajectory

2. Plan with 5s future trajectory

Activity!

Example: Learning forecasts for merging actors



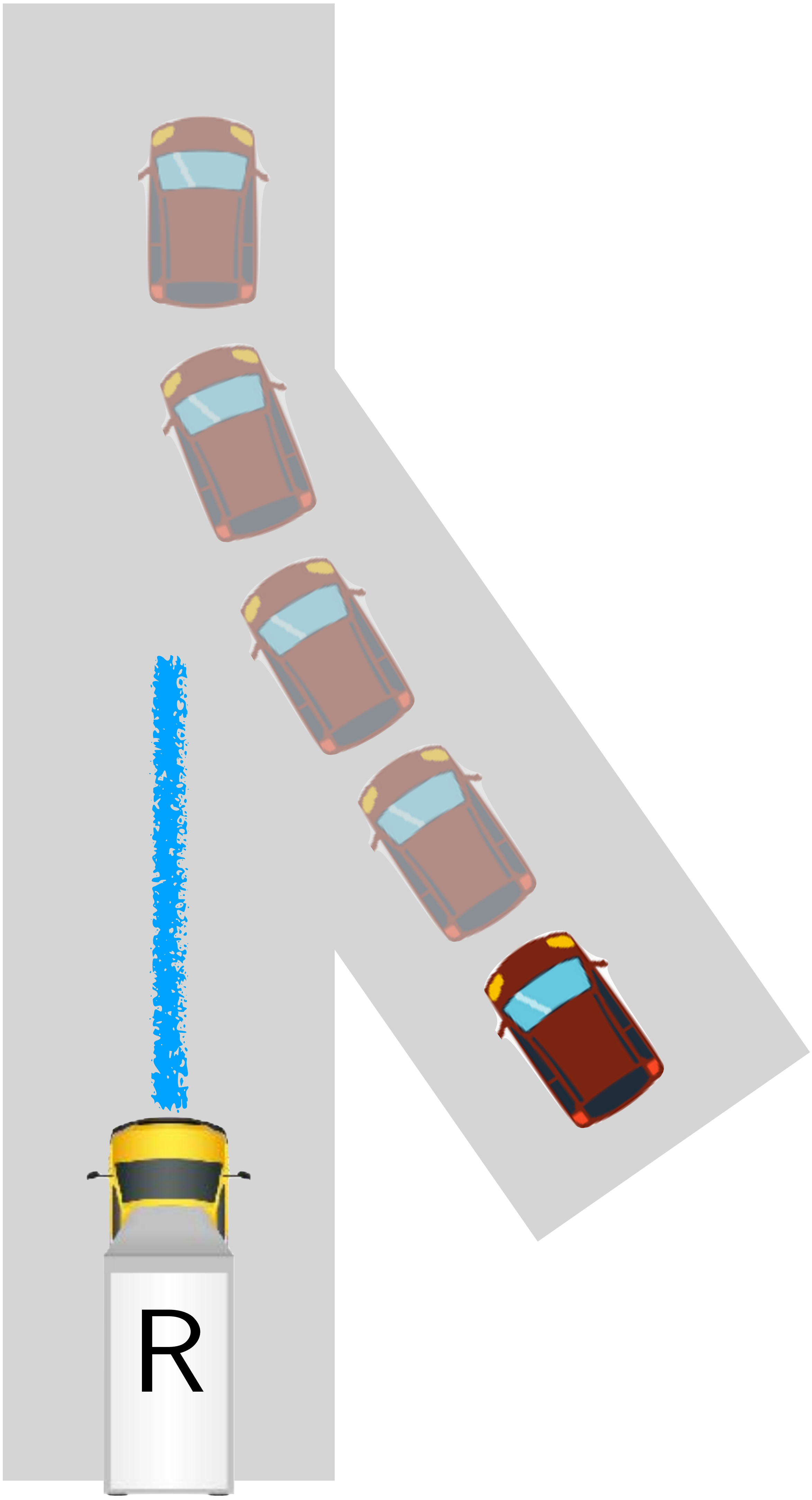
1. Predict 5s future trajectory

Data?

Model?

Loss?

Example: Learning forecasts for merging actors



2. Plan with 5s future trajectory

Cost function?

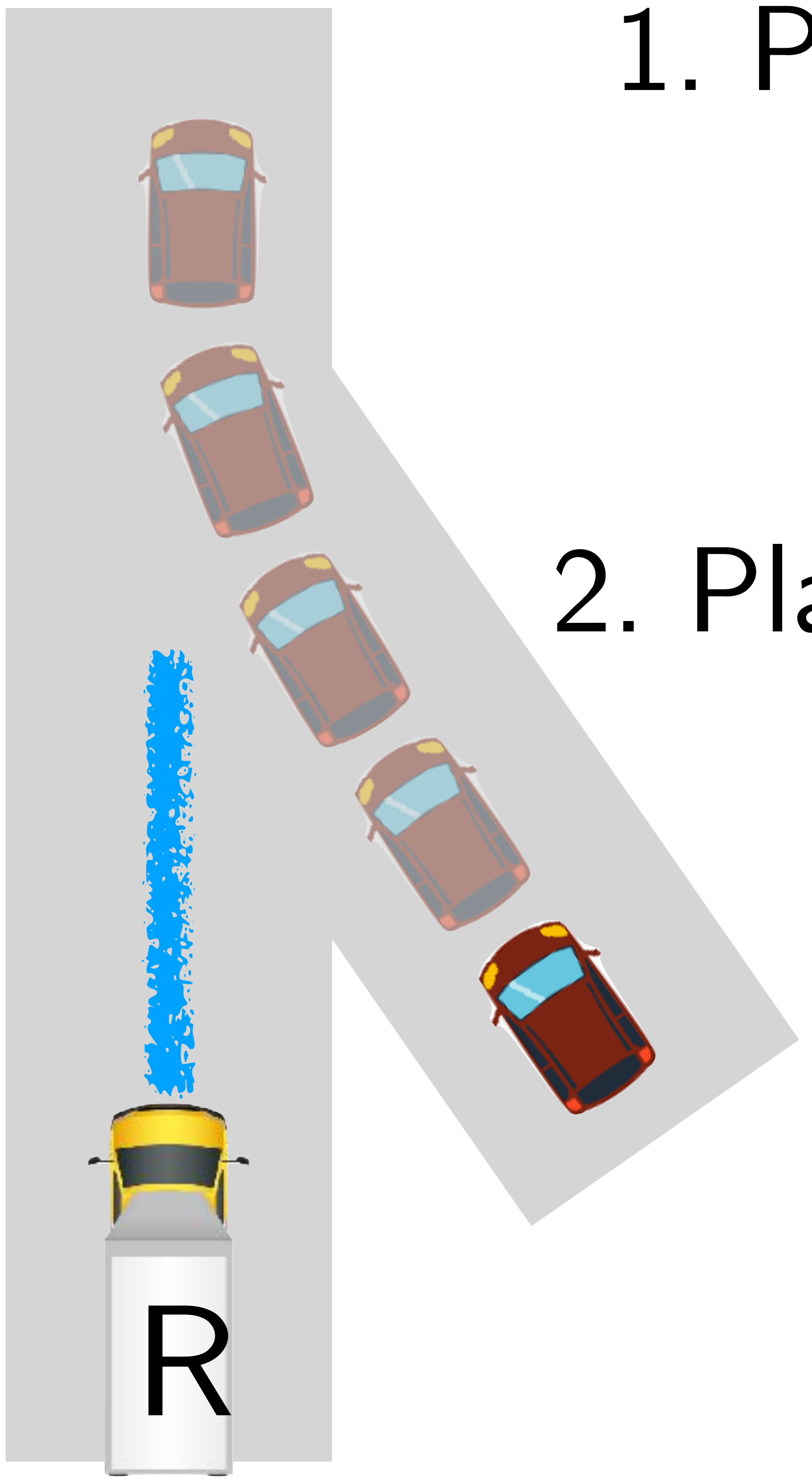
Planner?

Think-Pair-Share!

Think (30 sec): Design choices for forecasting and motion planning

Pair: Find a partner

Share (45 sec): Partners exchange ideas



1. Predict 5s future trajectory

Data? Model? Loss?

2. Plan with 5s future trajectory

Cost Function? Planner?

Why is current state insufficient to predict future?

Simple latent variables:

Velocity, Acceleration may not be observable

Complex latent variables:

Intent (turning left, making a lane change) are
not observable and must be inferred from past actions

A very brief history of sequence prediction in robotics

Kalman Filter + Prediction

Hand design observation models, infer latent states, forward predict.

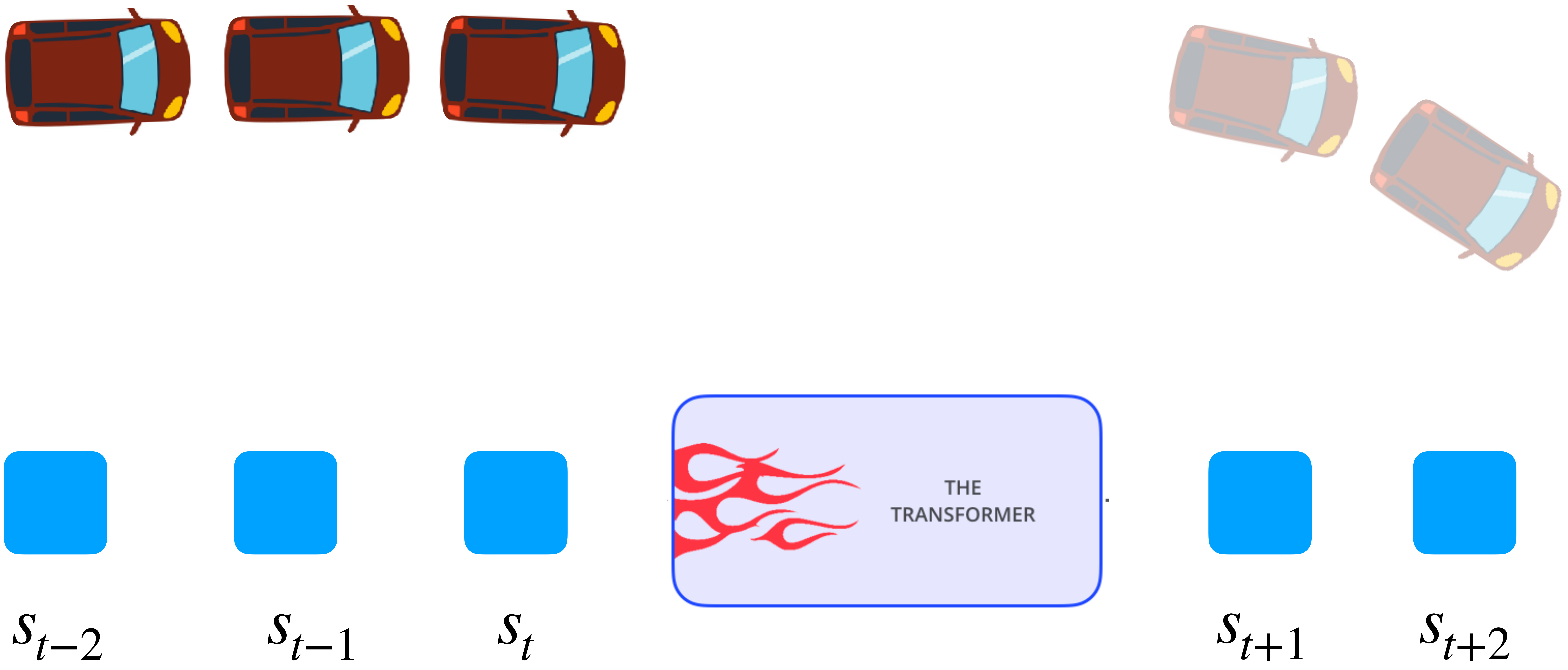
RNN, LSTMs

Learn the filter! Problem - forget long sequences since only one hidden state vector passed from one time step to next

Transformers

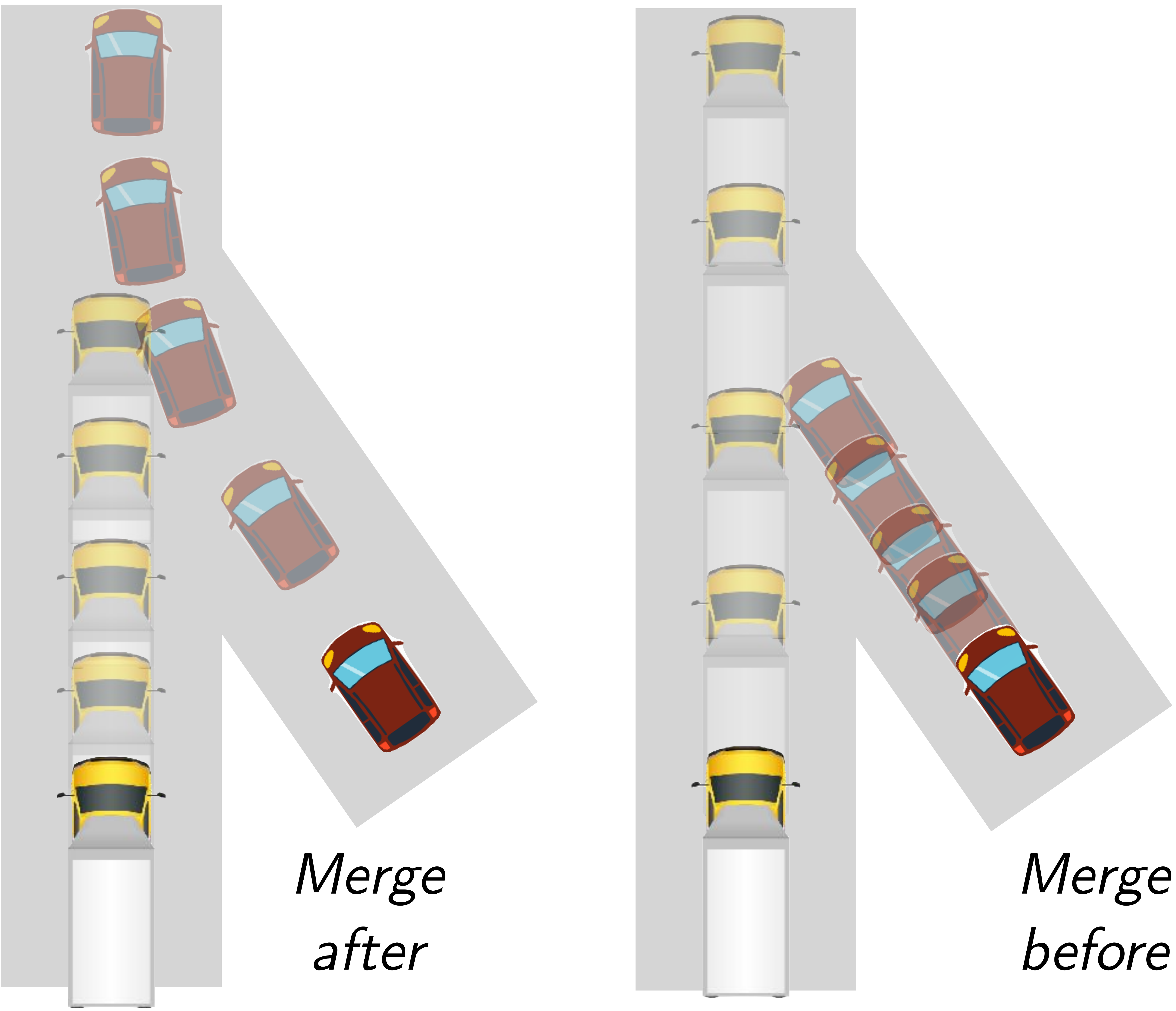
Retain all hidden state, pay O(H^2) computation

Model: Use a transformer to map history to future

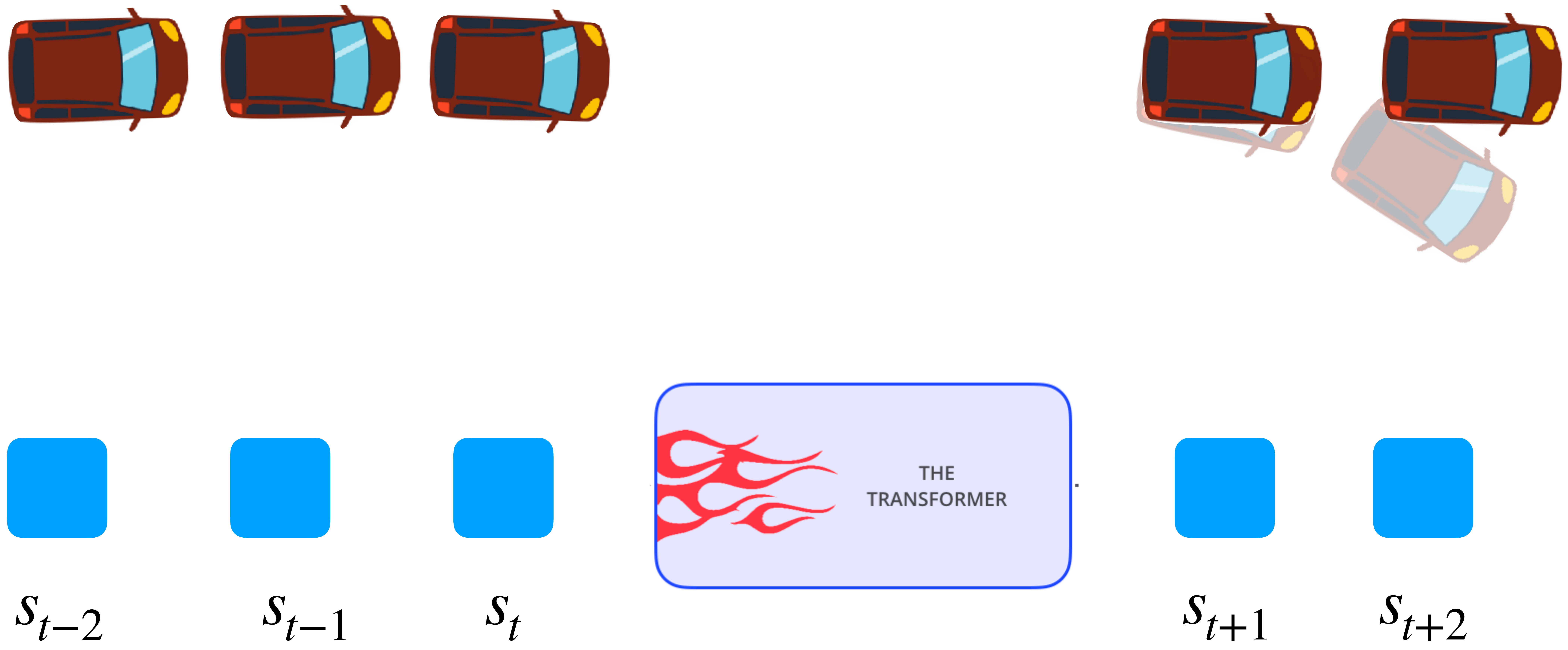


Data: Drive around the car and collect data

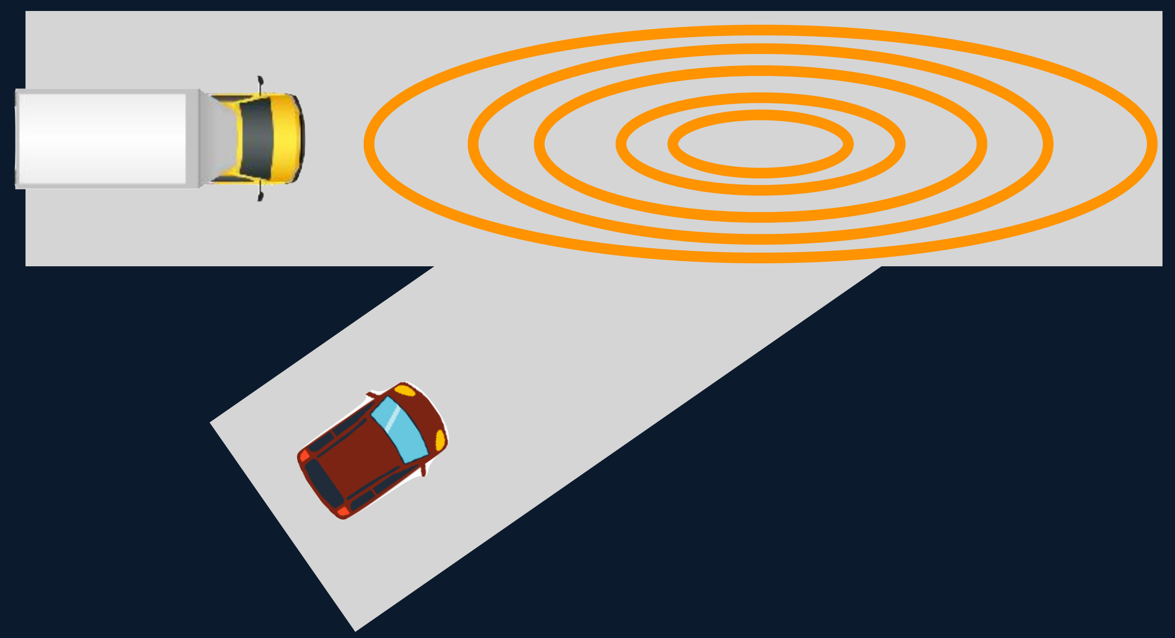
Train Data



Loss: L2 Loss from Ground Truth



We have model, data, loss.
Let's deploy!



Forecasts have huge variance!

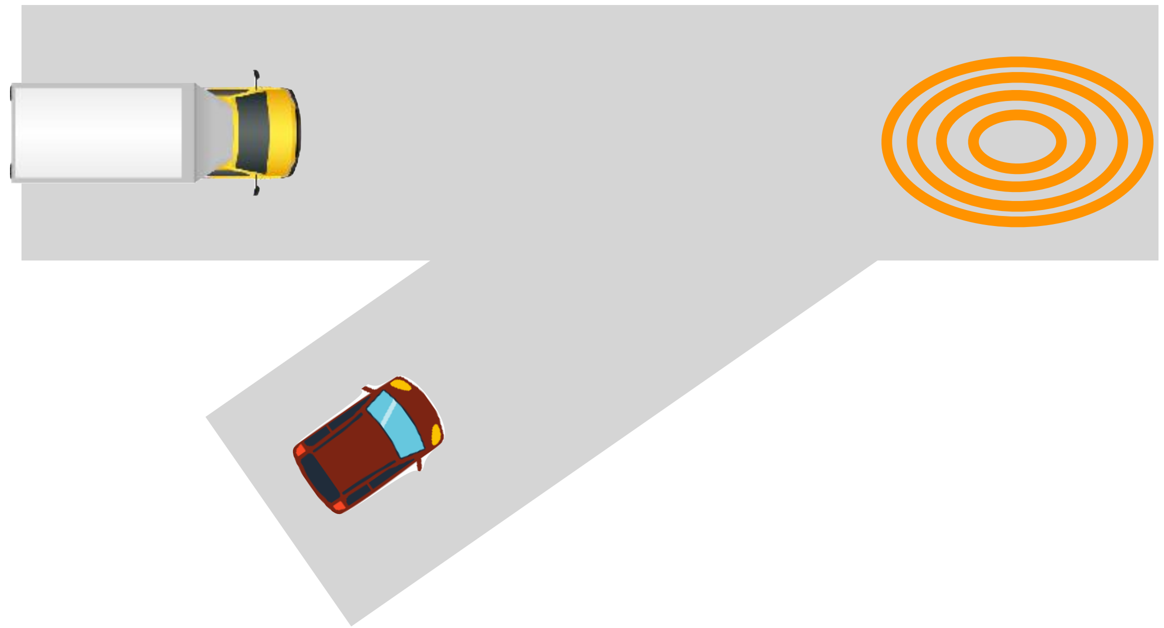
Planner brakes aggressively!

Why is the forecast so whacky?

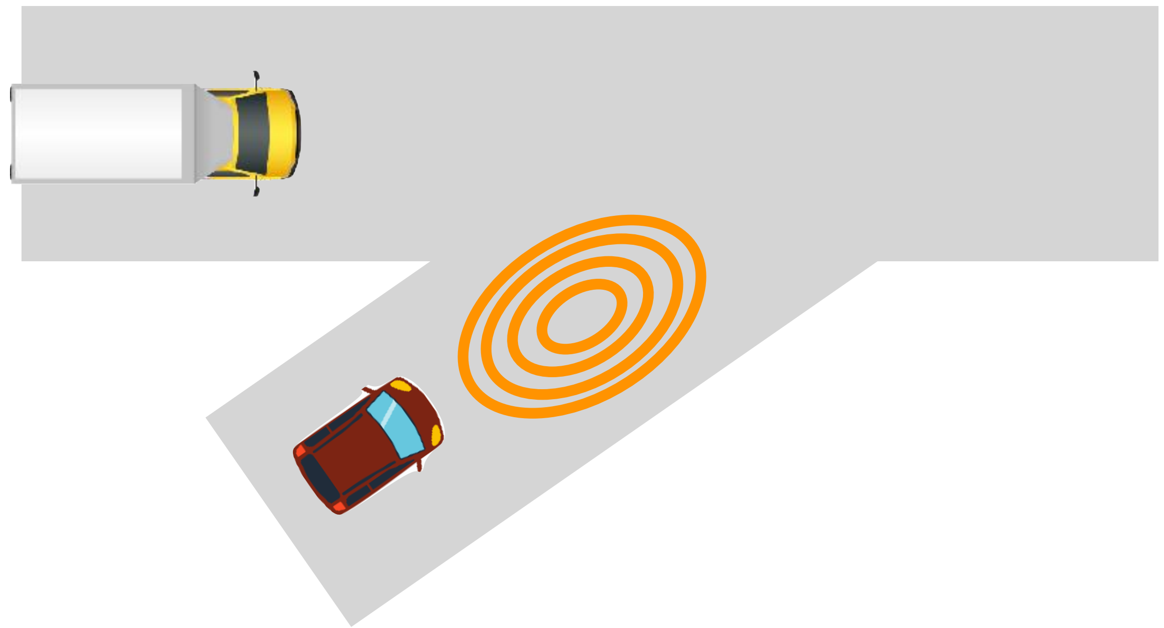
Why is the forecast so whacky?

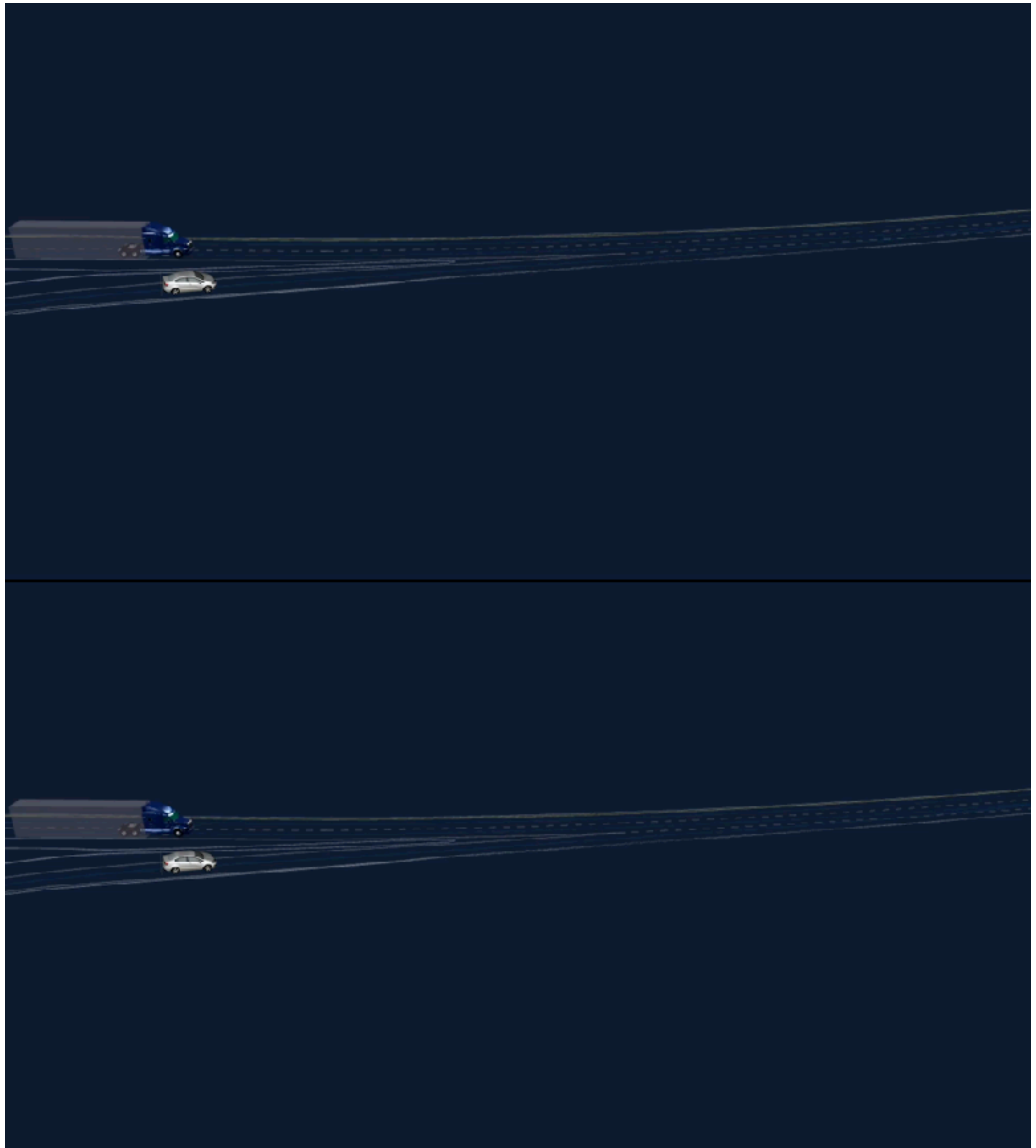
Marginalizing over multiple modes!

Mode A:
Robot merges after



Mode B:
Robot merges before



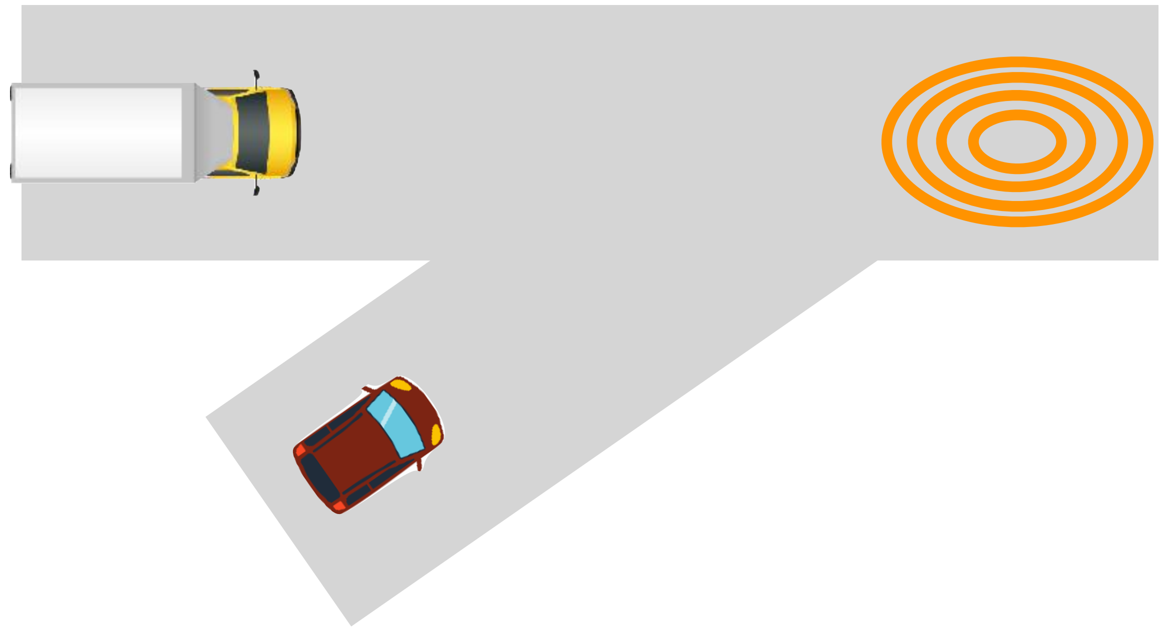


Okay .. so why can't we just predict multi-modal distributions?

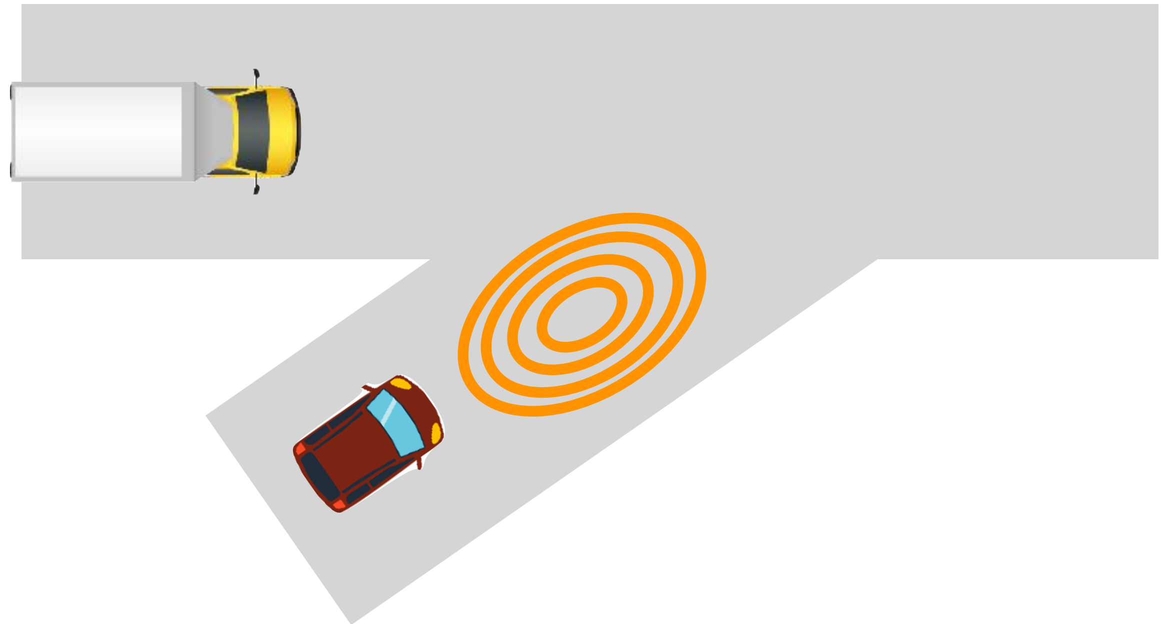
Multi-modal forecasts do not solve the issue!

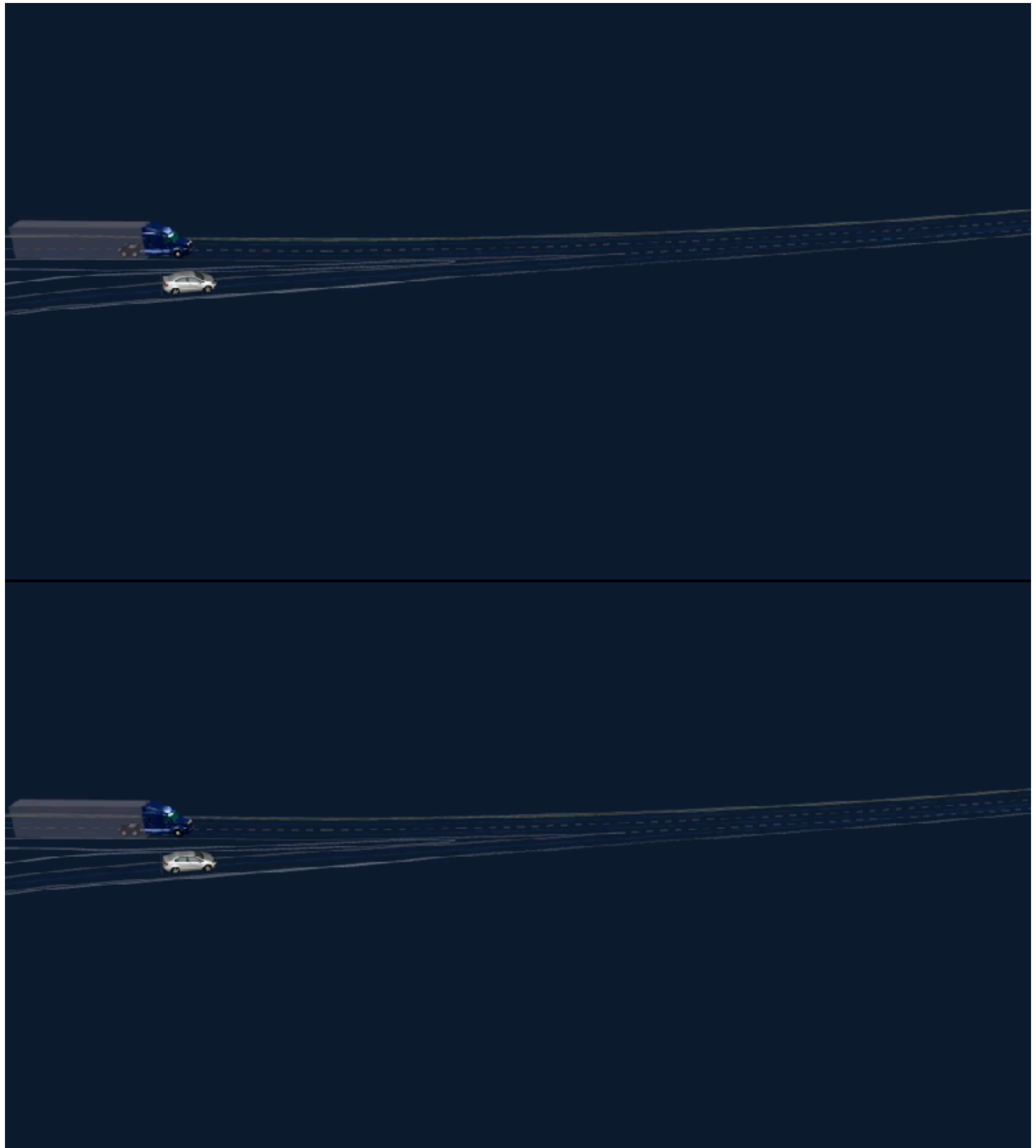
We are (incorrectly) telling the planner both modes can happen!

Mode A:
Robot merges after



Mode B:
Robot merges before

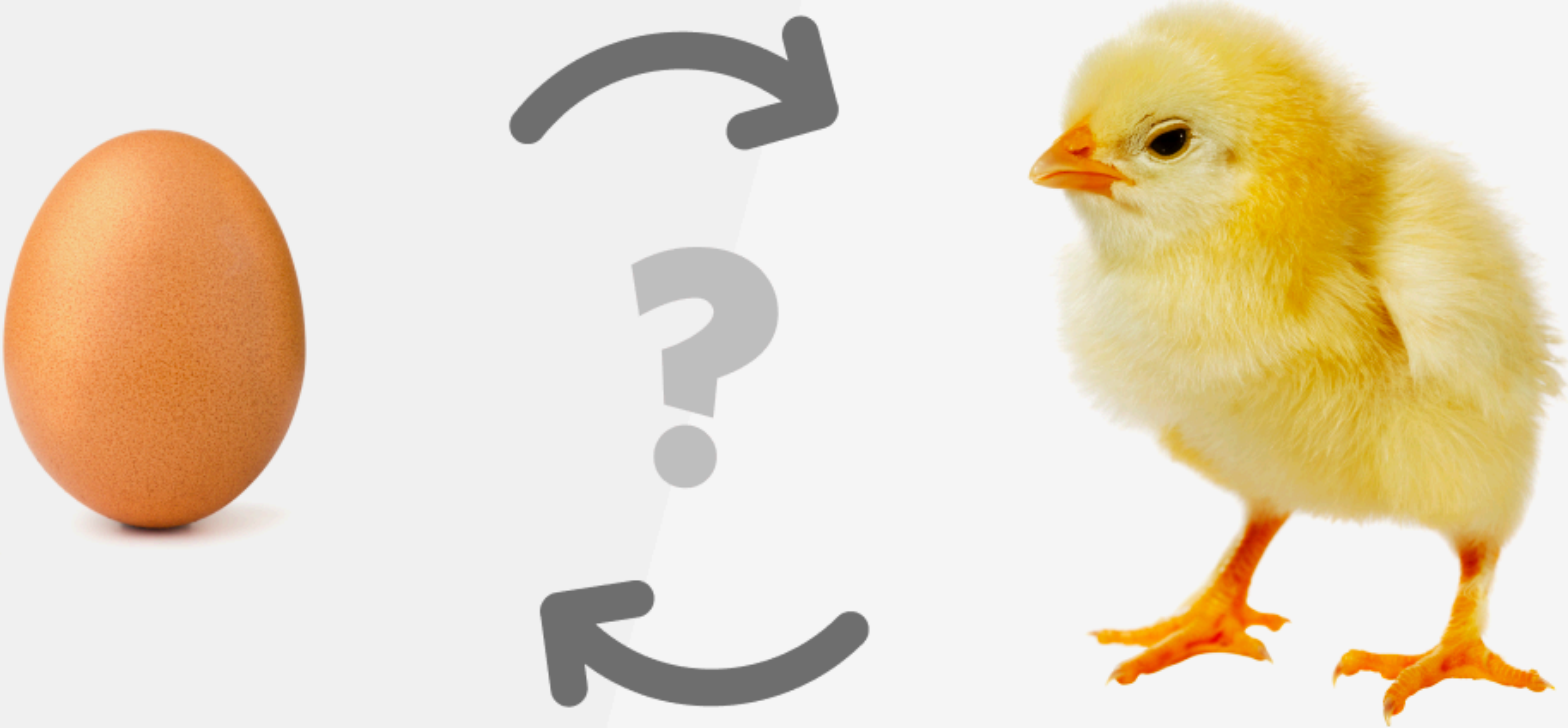




What robot does depends on other humans

What other humans do depends on the robot

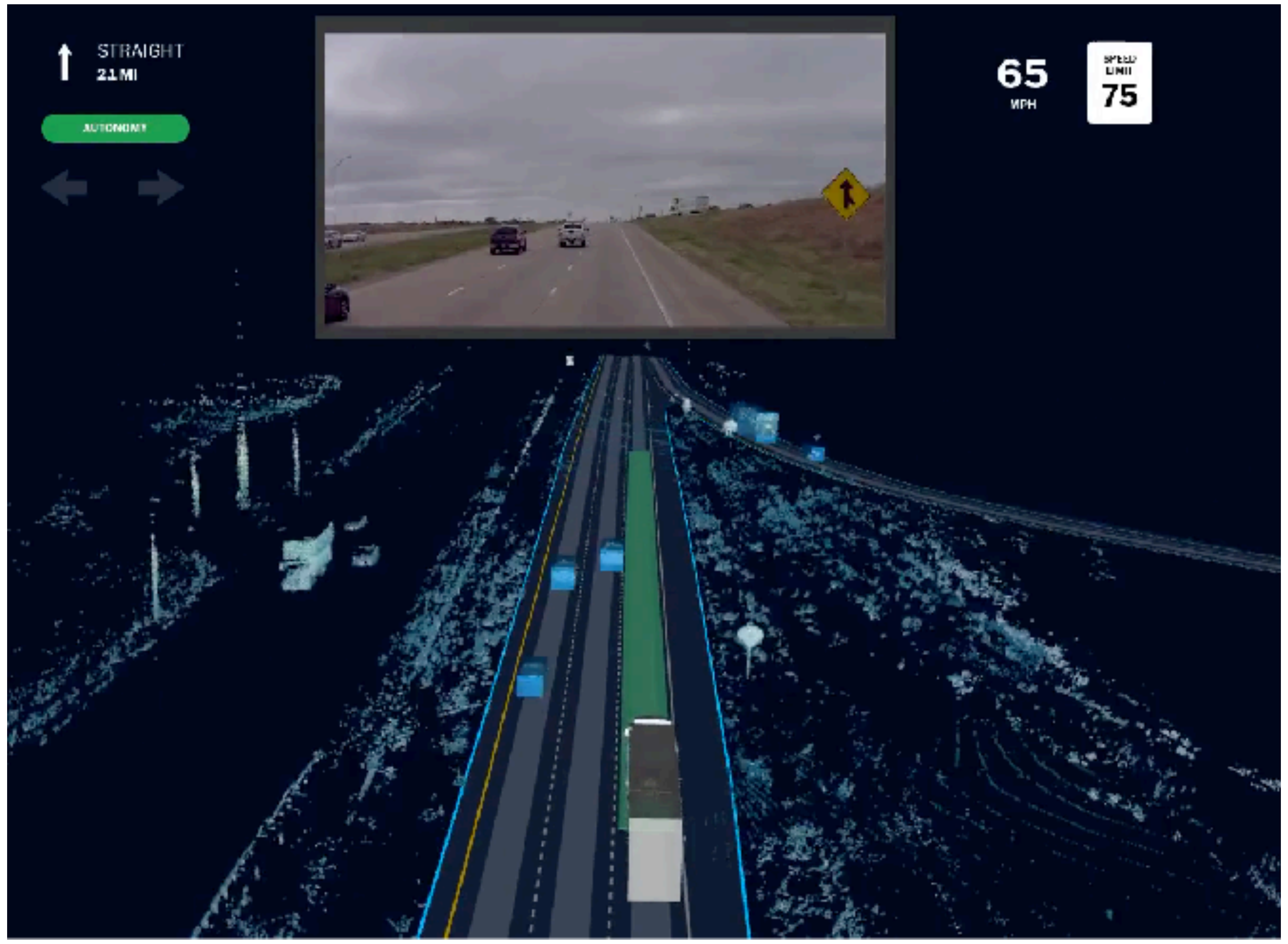
Forecasting-or-planning:
a chicken-or-egg problem



Why can't we just forecast the robot motion?

Planning is NOT merely forecasting

Suppose you collected data from this



vs data from this

Which data is useful for forecasting? For imitation learning?

Solving the chicken-or-egg problem

Train a *conditional* forecasting model

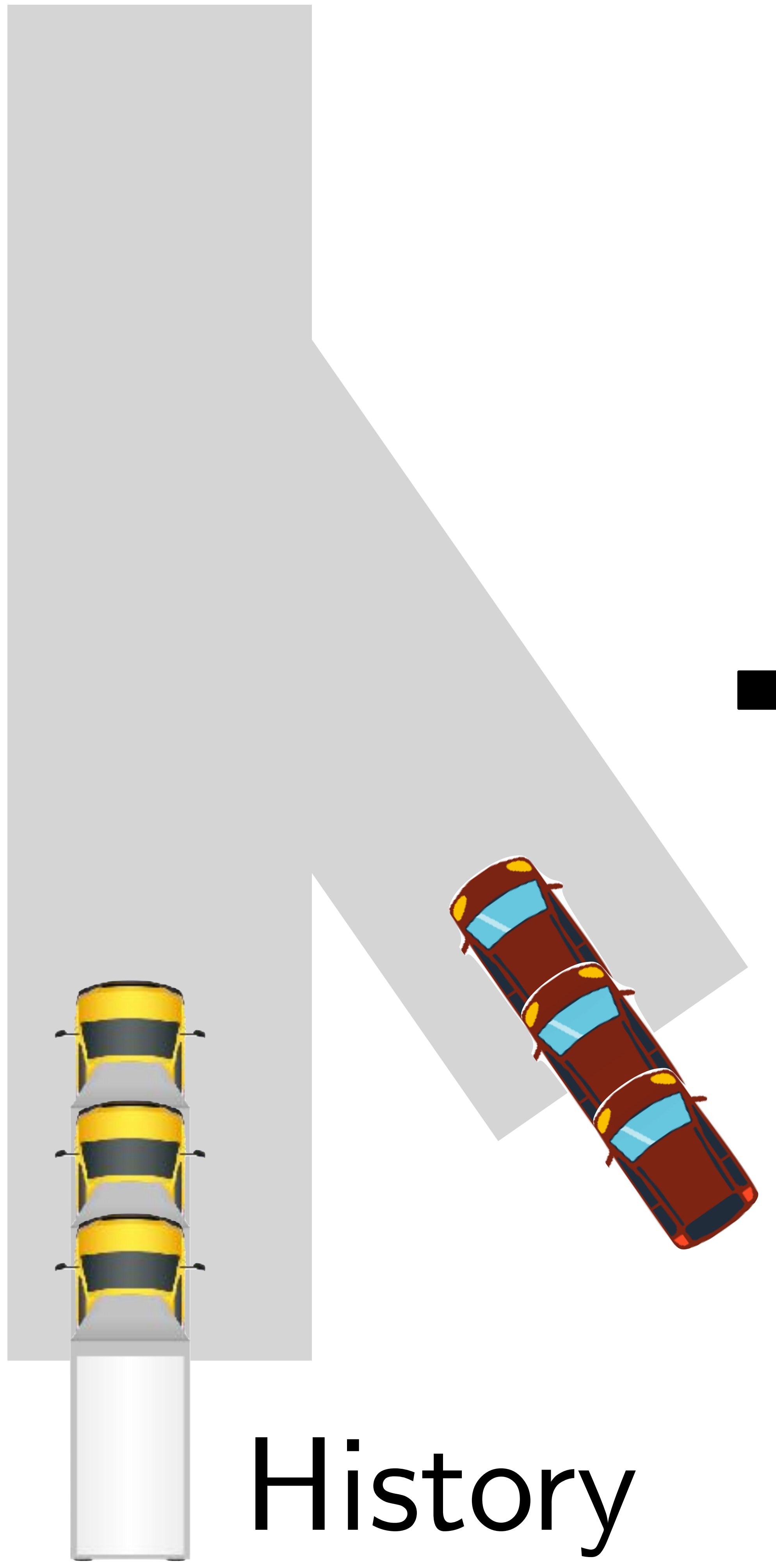
Marginal forecasting

$P(s_{t:t+k} \mid s_{t:t-k})$

Conditional forecasting

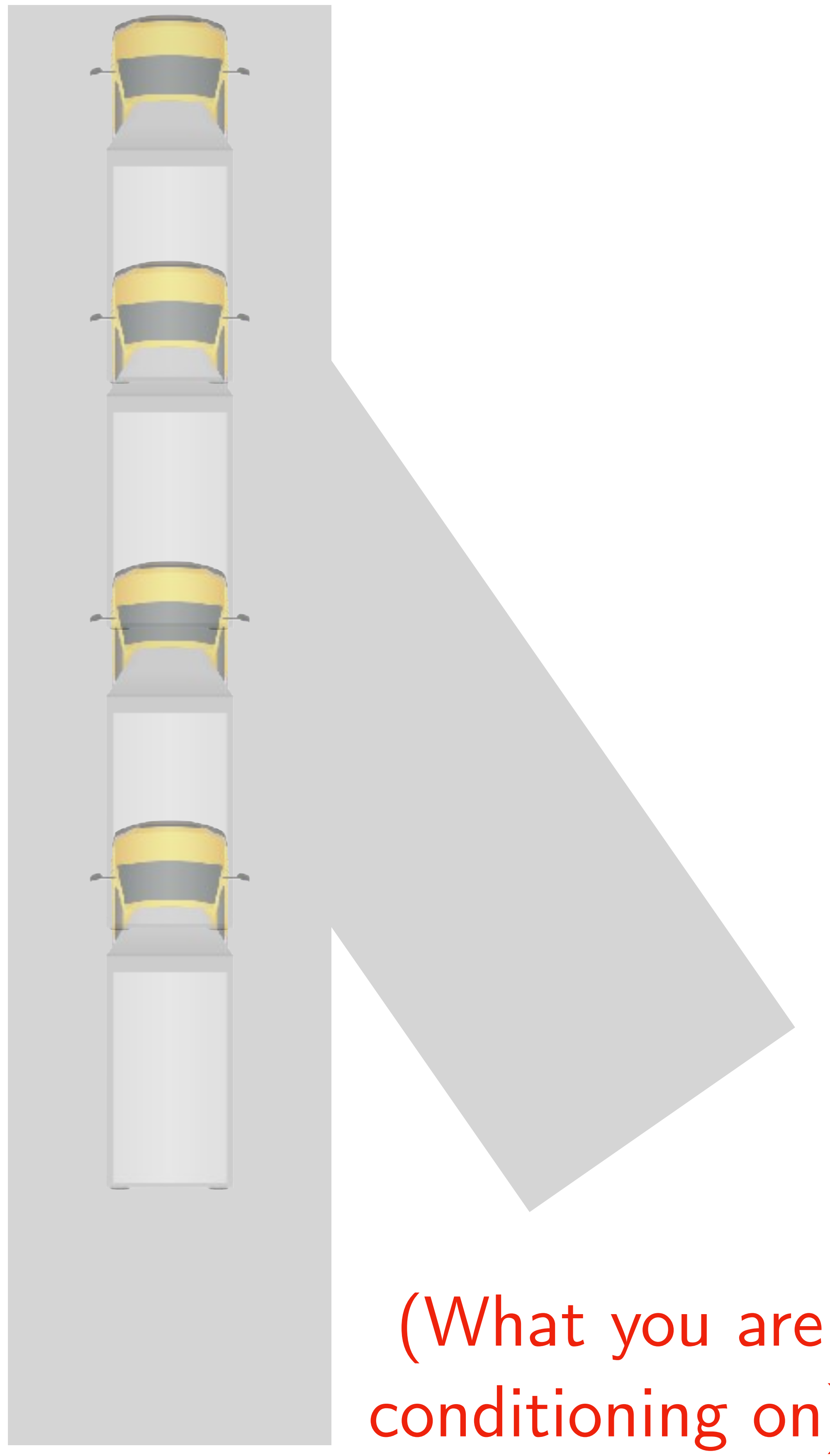
$P(s_{t:t+k} \mid s_{t:t-k}, \xi_{plan})$

Solution: Train Conditional Forecasts



History

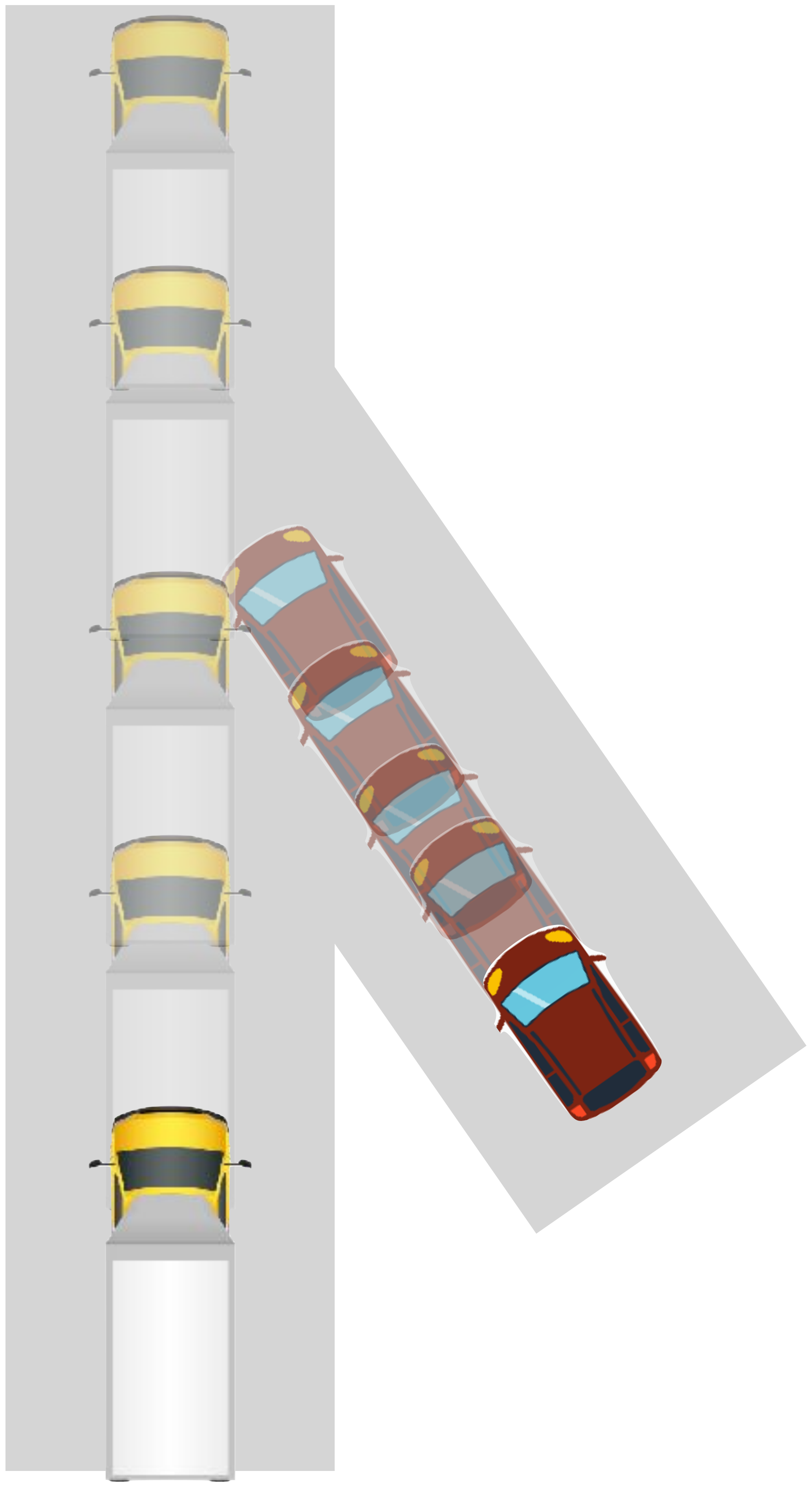
$\ldots, s_{t-2}, s_{t-1}, s_t$



(What you are conditioning on)

Plan

$a_{t_1}, a_{t+2}, a_{t+3}$

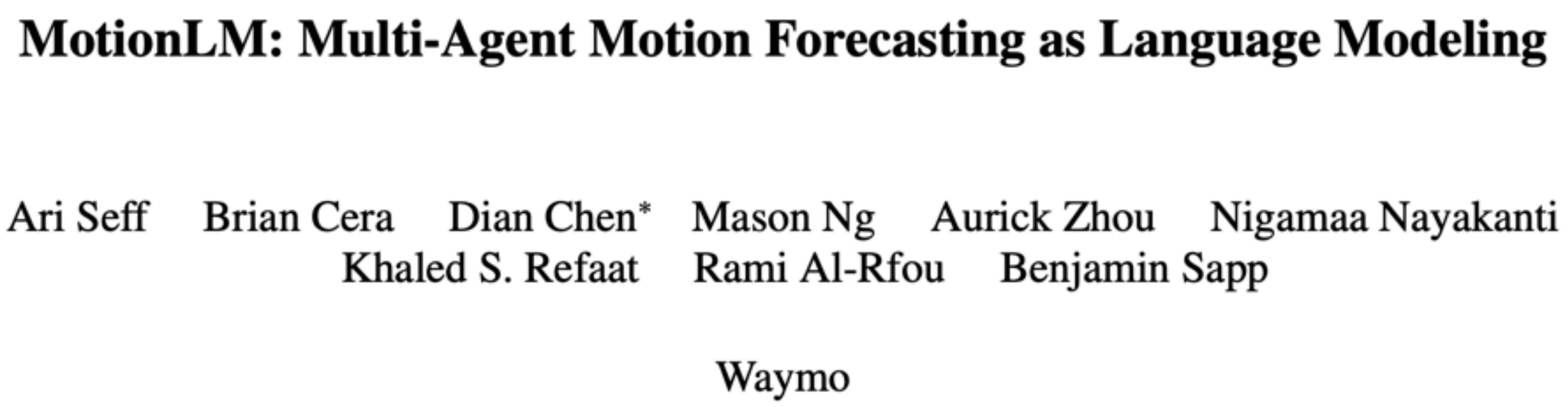


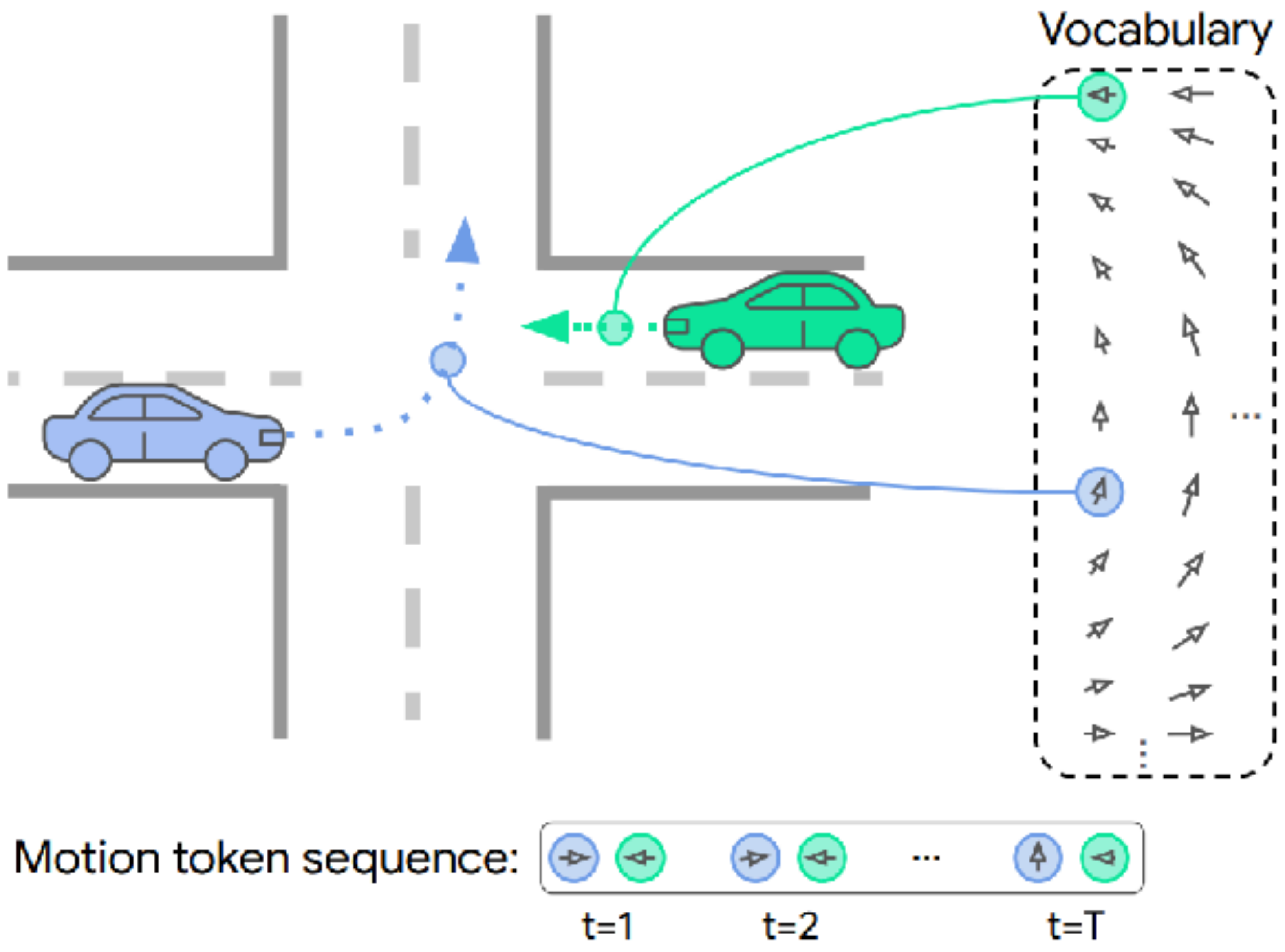
Forecast

$s_{t+1}, s_{t+2}, s_{t+3}, s_{t+4}$

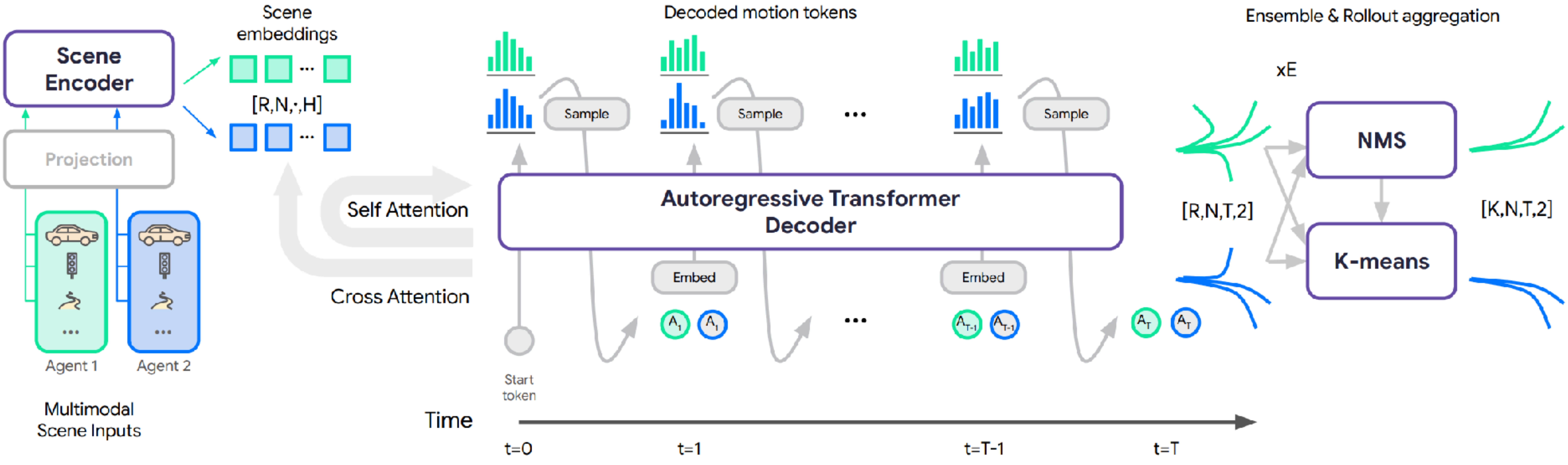
How do we do this?

Language Models for Forecasting

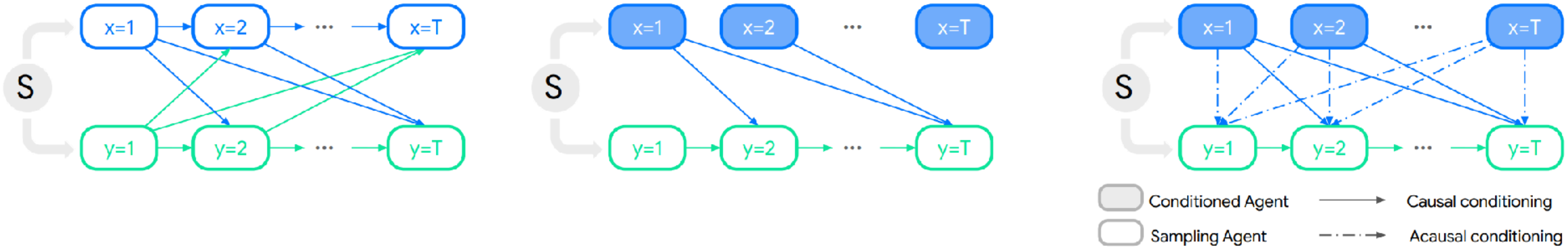




Architecture



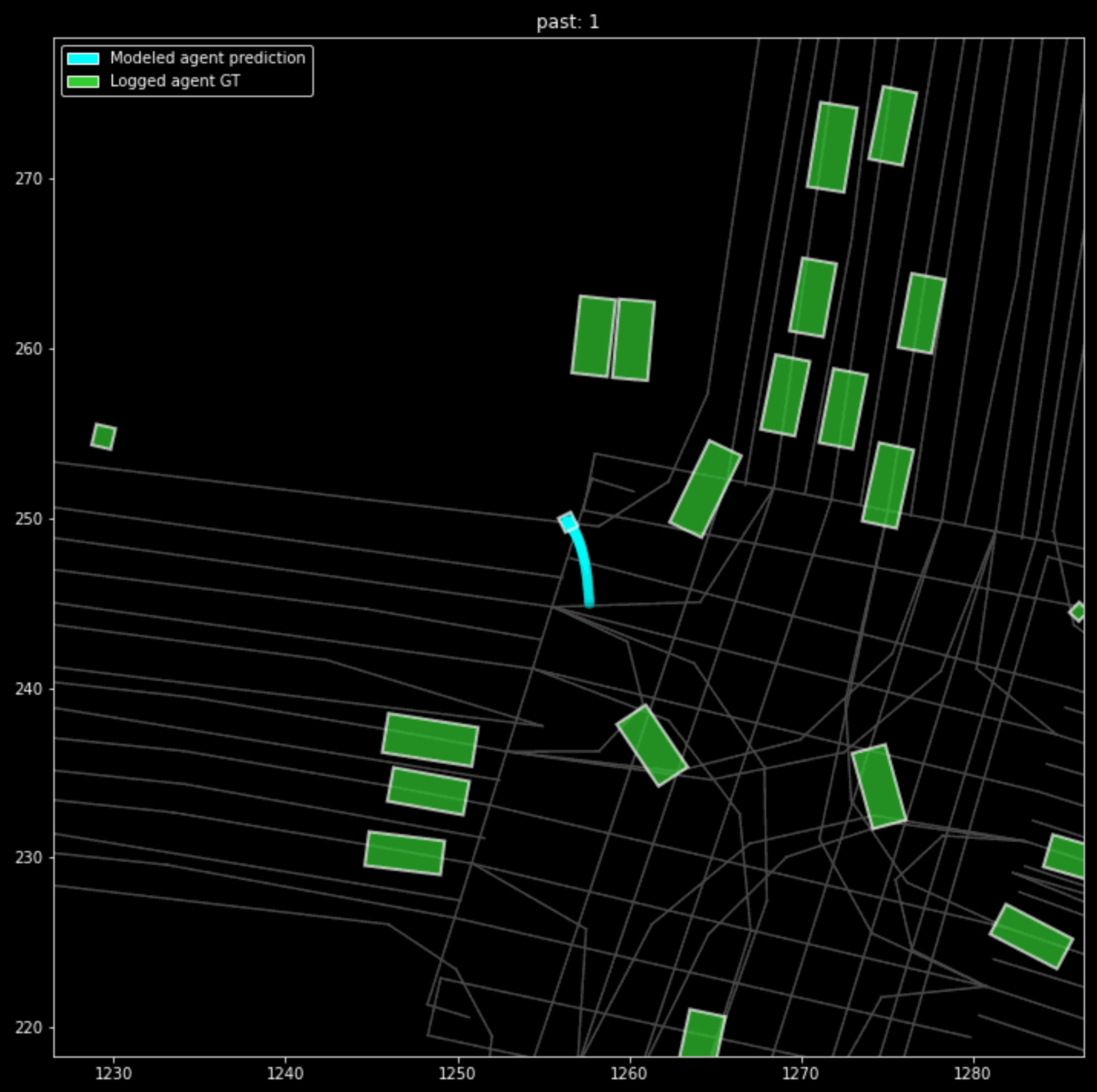
How can we condition on the robot future?

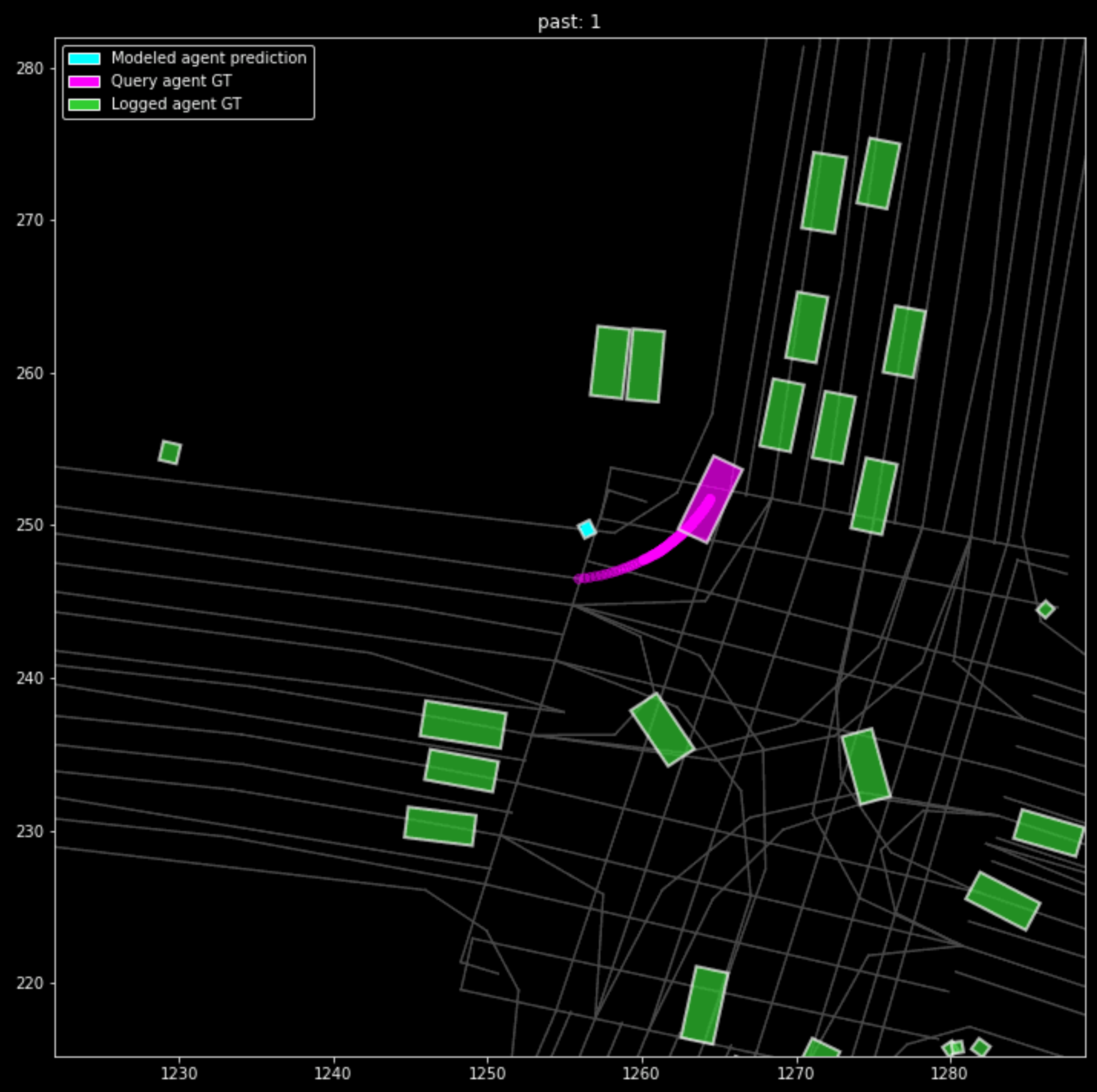


No future conditioning:
Causal Attention

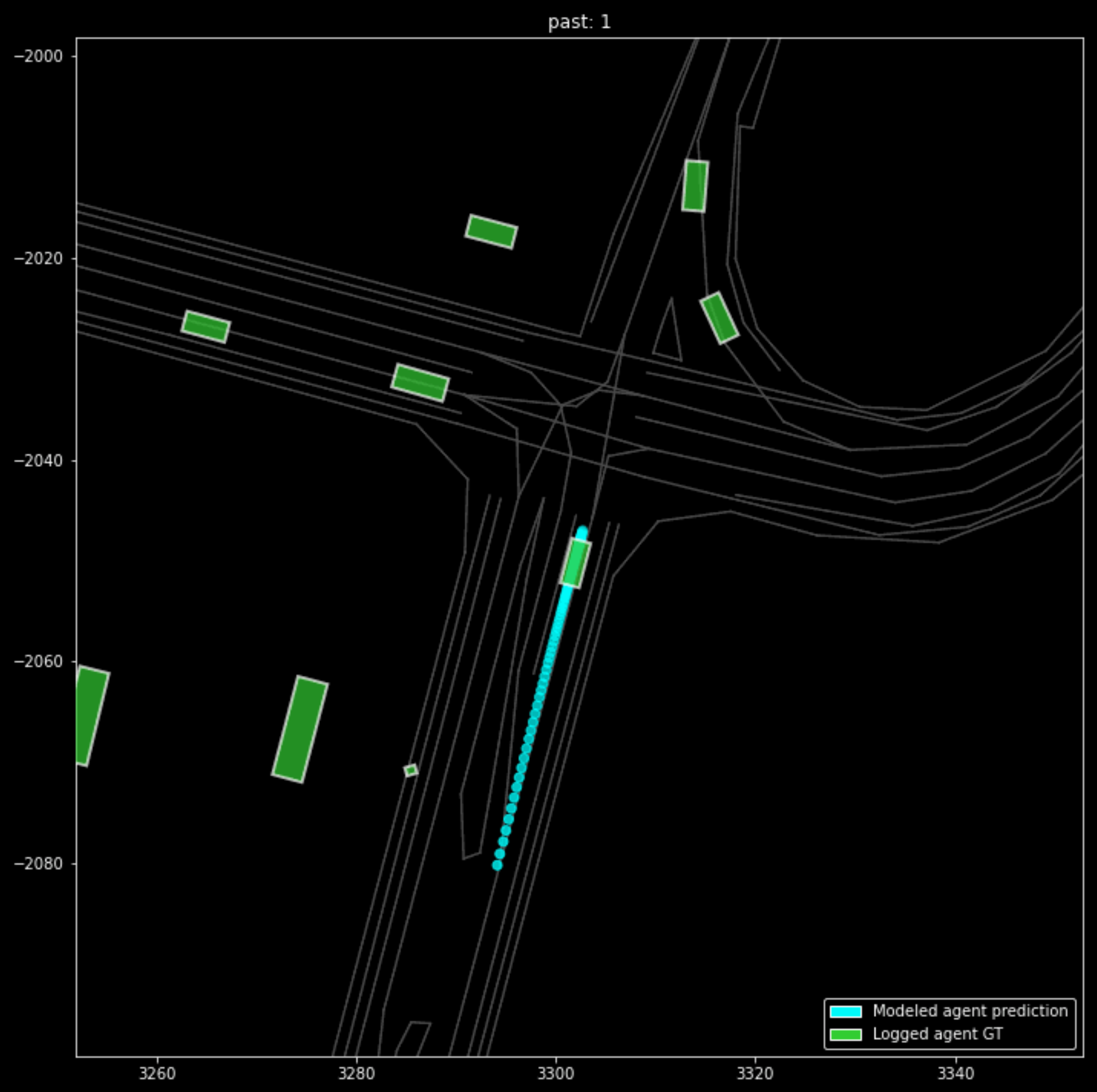
Future conditioning:
Bi-Directional Attention

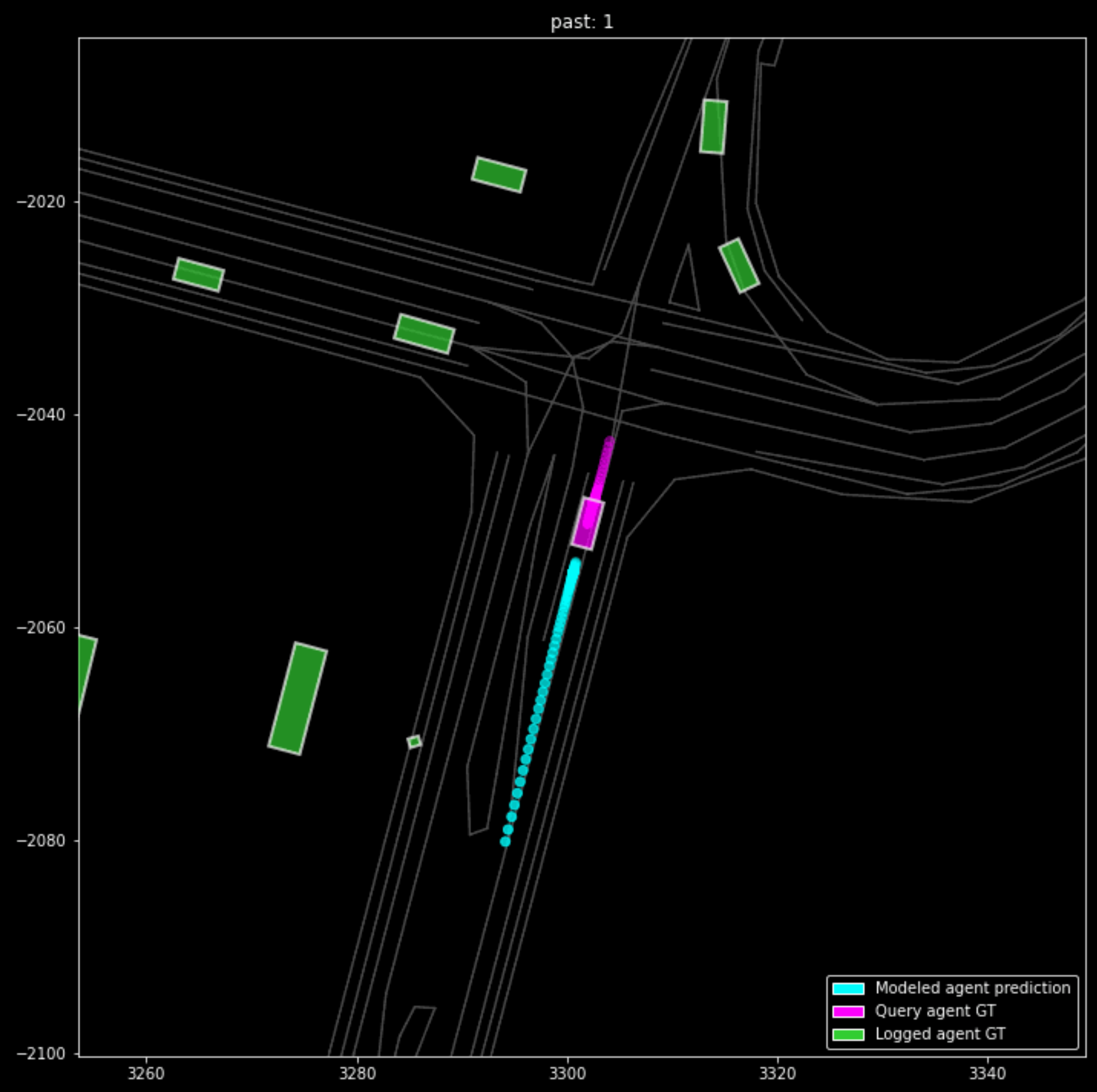
Marginal vs Conditional





Marginal vs Conditional





How can I use conditional forecasts in practice?

Pseudo code for planning with forecasts

Initialize with a library of candidate trajectories Ξ

For $\xi_{plan} \in \Xi$:

Call conditional forecast with history and ξ_{plan} to predict $\xi_{forecast}$ for all the agents

Compute cost of ξ_{plan} using $\xi_{forecast}$

Return cheapest plan ξ^*_{plan}

Pseudo code for planning with forecasts

Initialize with a library of candidate trajectories Ξ

For $\xi_{plan} \in \Xi$:

Call conditional forecast with history and ξ_{plan} to predict $\xi_{forecast}$ for all the agents

(Can do this in a batch!)

Compute cost of ξ_{plan} using $\xi_{forecast}$

Return cheapest plan ξ^*_{plan}

Pseudo code for planning with forecasts

Initialize with a library of candidate trajectories Ξ

For $\xi_{plan} \in \Xi$:

  Call conditional forecast with history and ξ_{plan} to predict $\xi_{forecast}$ for all the agents

  Compute cost of ξ_{plan} using $\xi_{forecast}$

Return cheapest plan ξ^*_{plan}

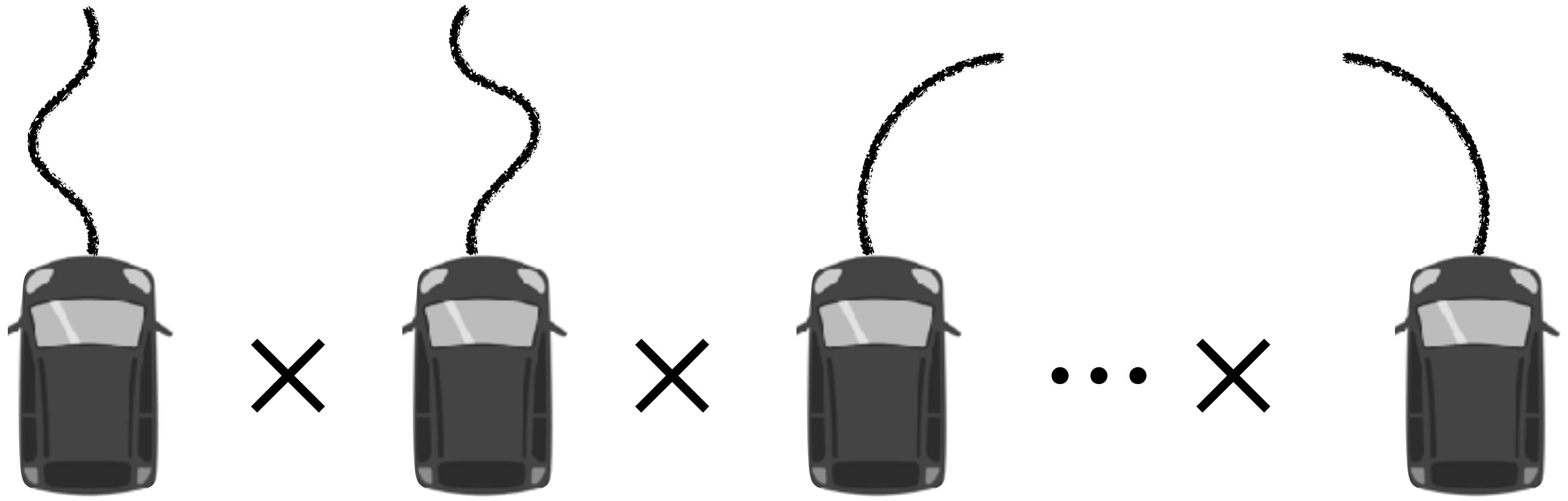
Trajectories are continuous sequences of motion. Space of all candidate trajectories is huge!!

Problem: Space of joint trajectories is massive

Continuous space of trajectories

+

Exponentially with in actors

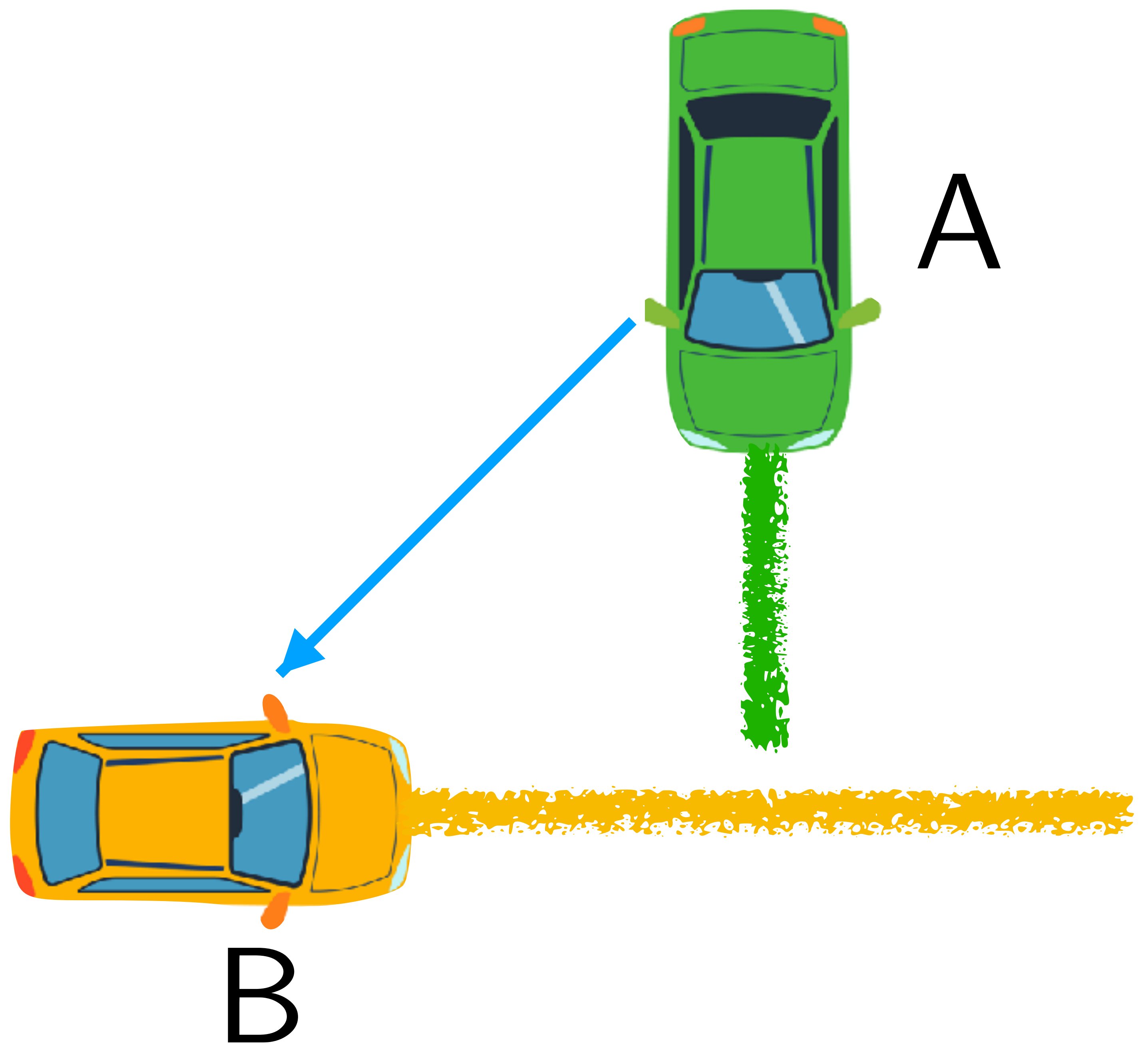


Conditional forecasting just makes this even harder

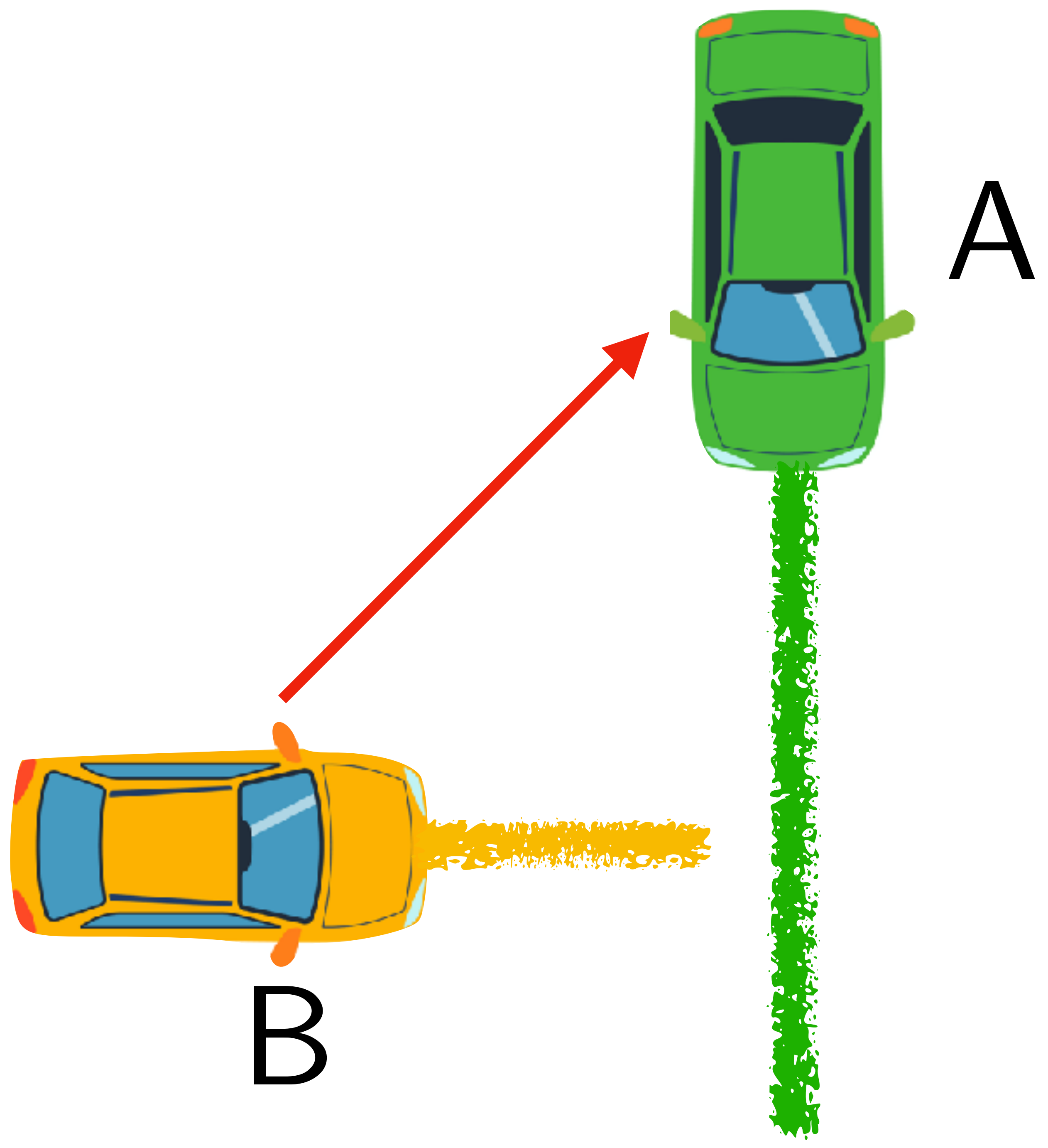
There is a discrete grammar for self-driving ...

3 fundamental modes of space-time paths

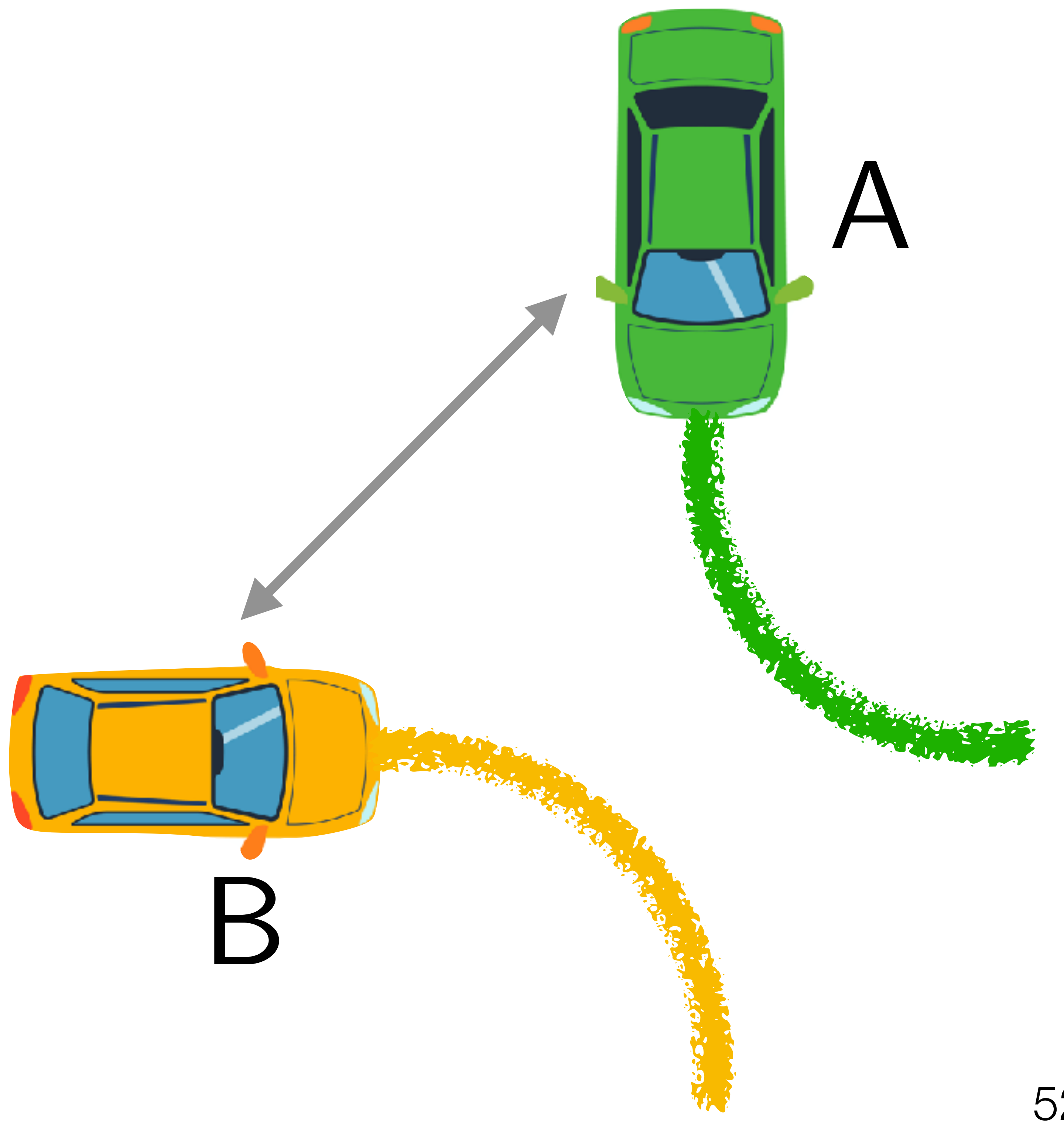
A Yields to B



B Yields to A



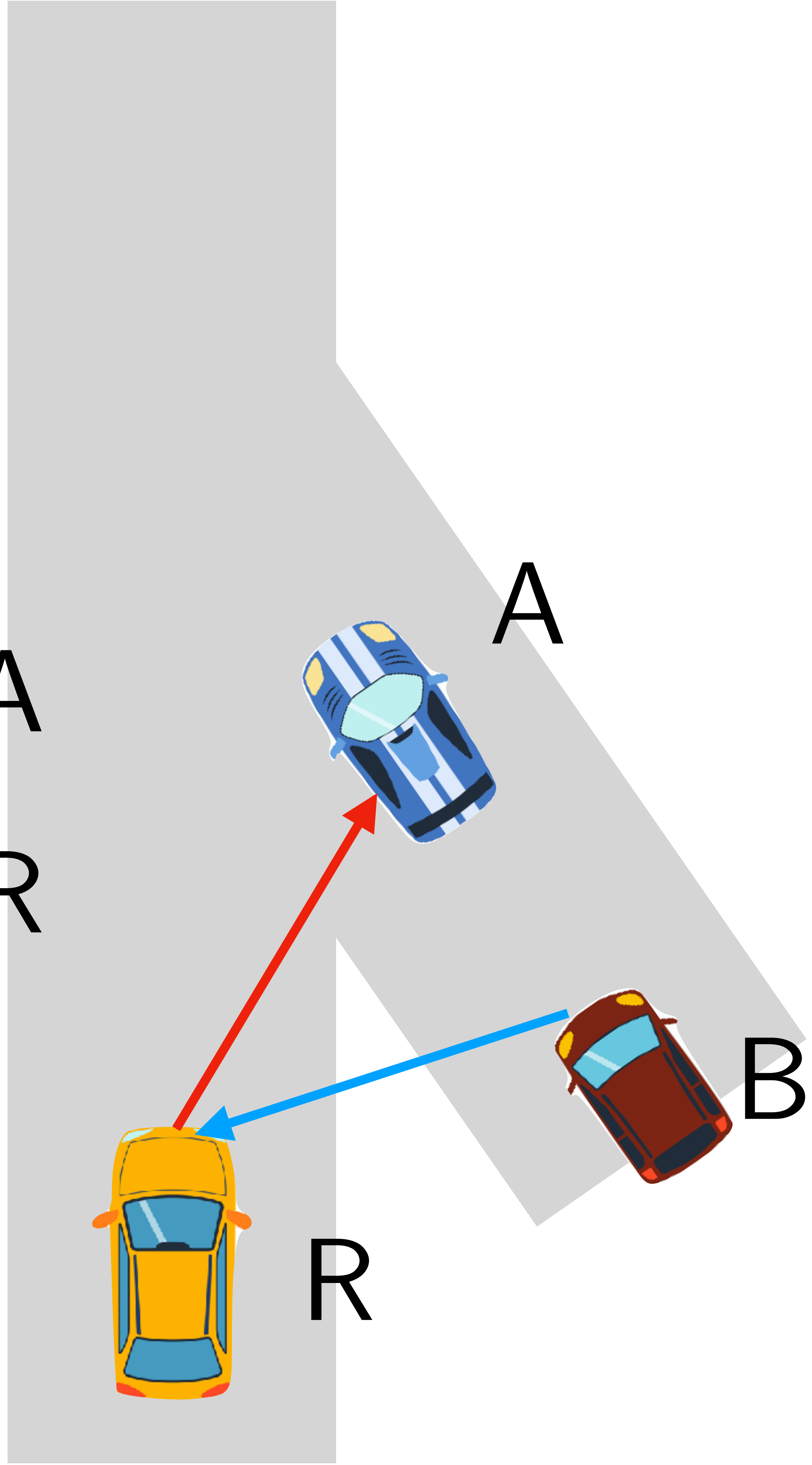
Not Yield



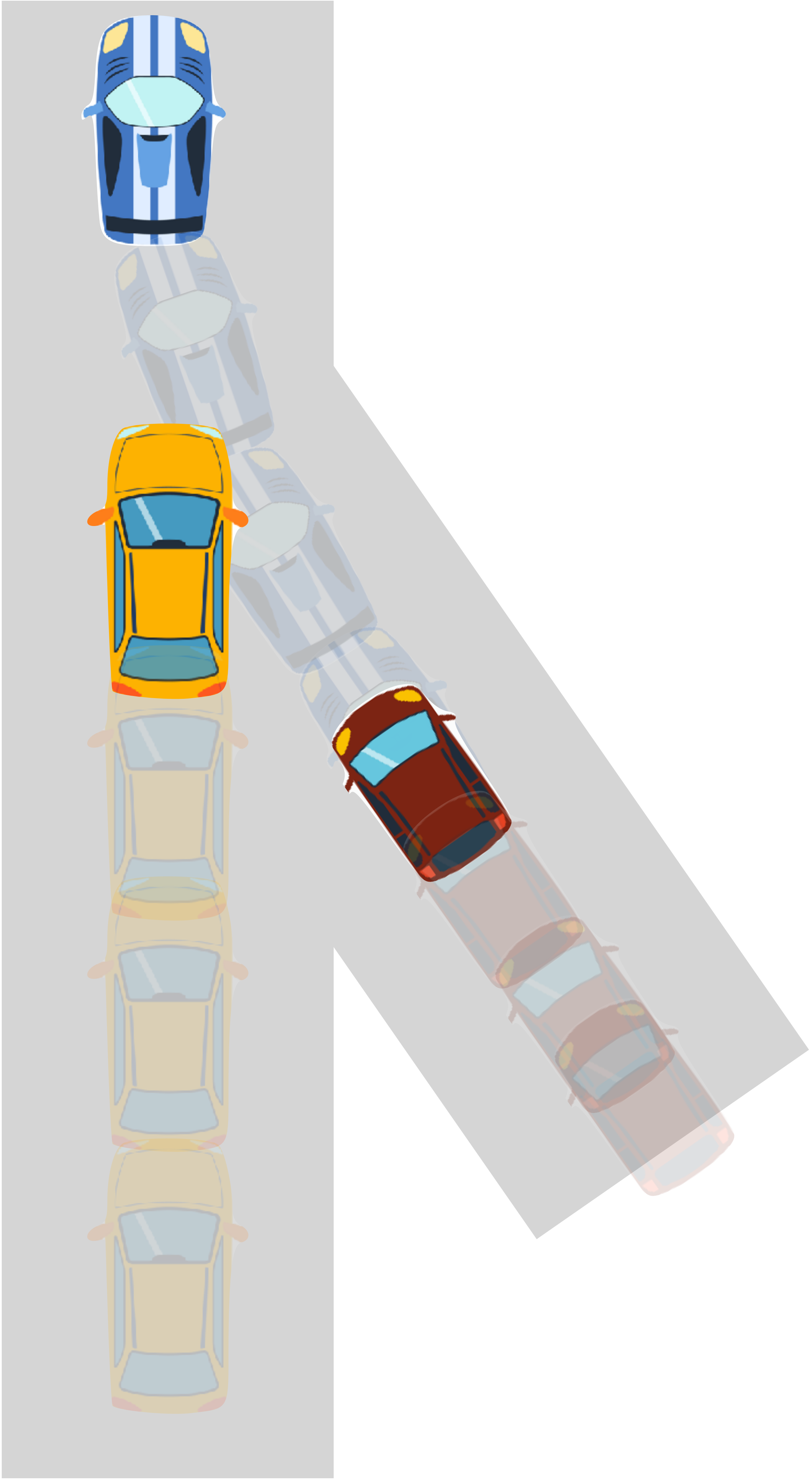
Mode ≡ A single basin of forecast

R Yields to A

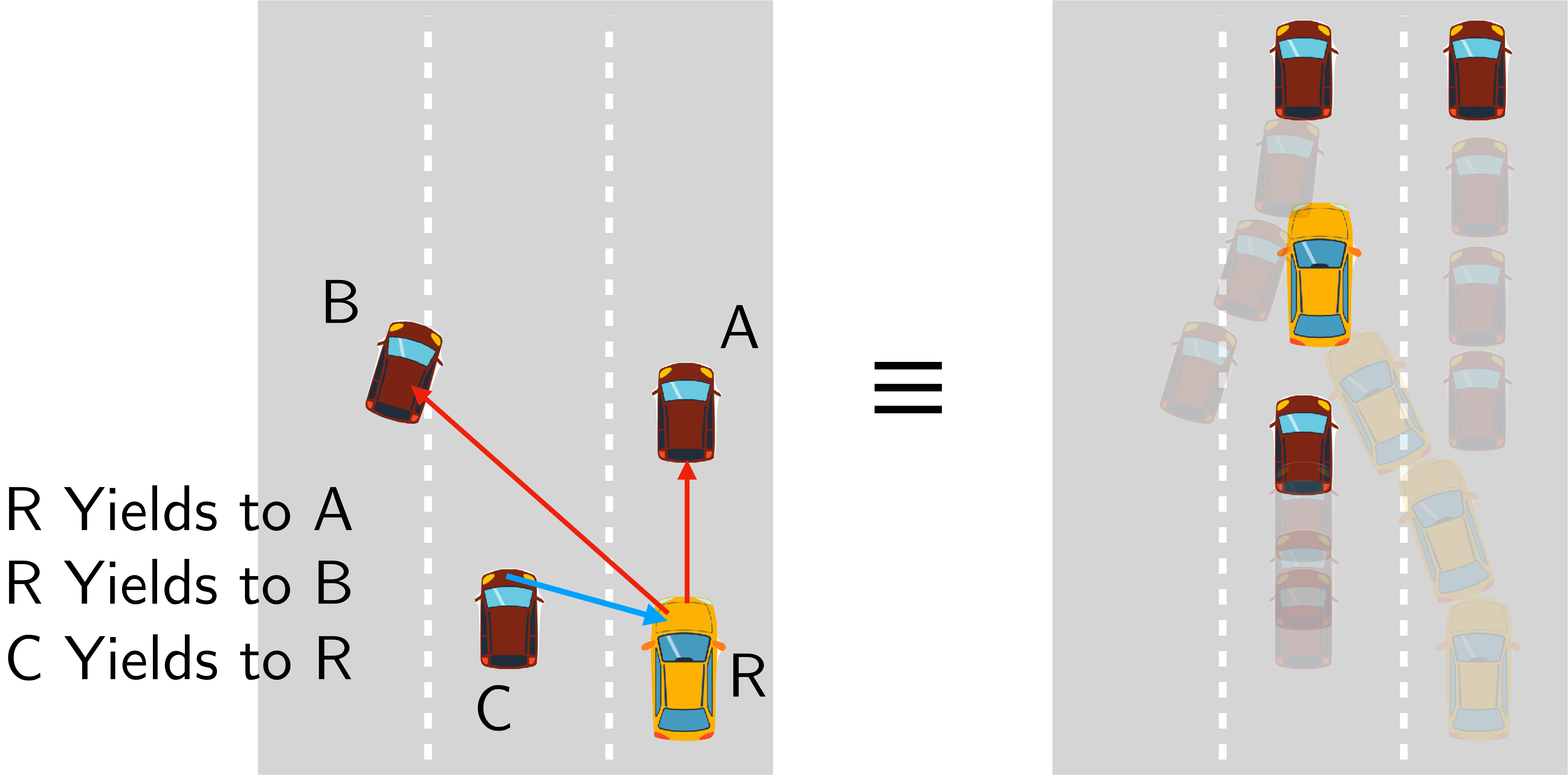
B Yields to R



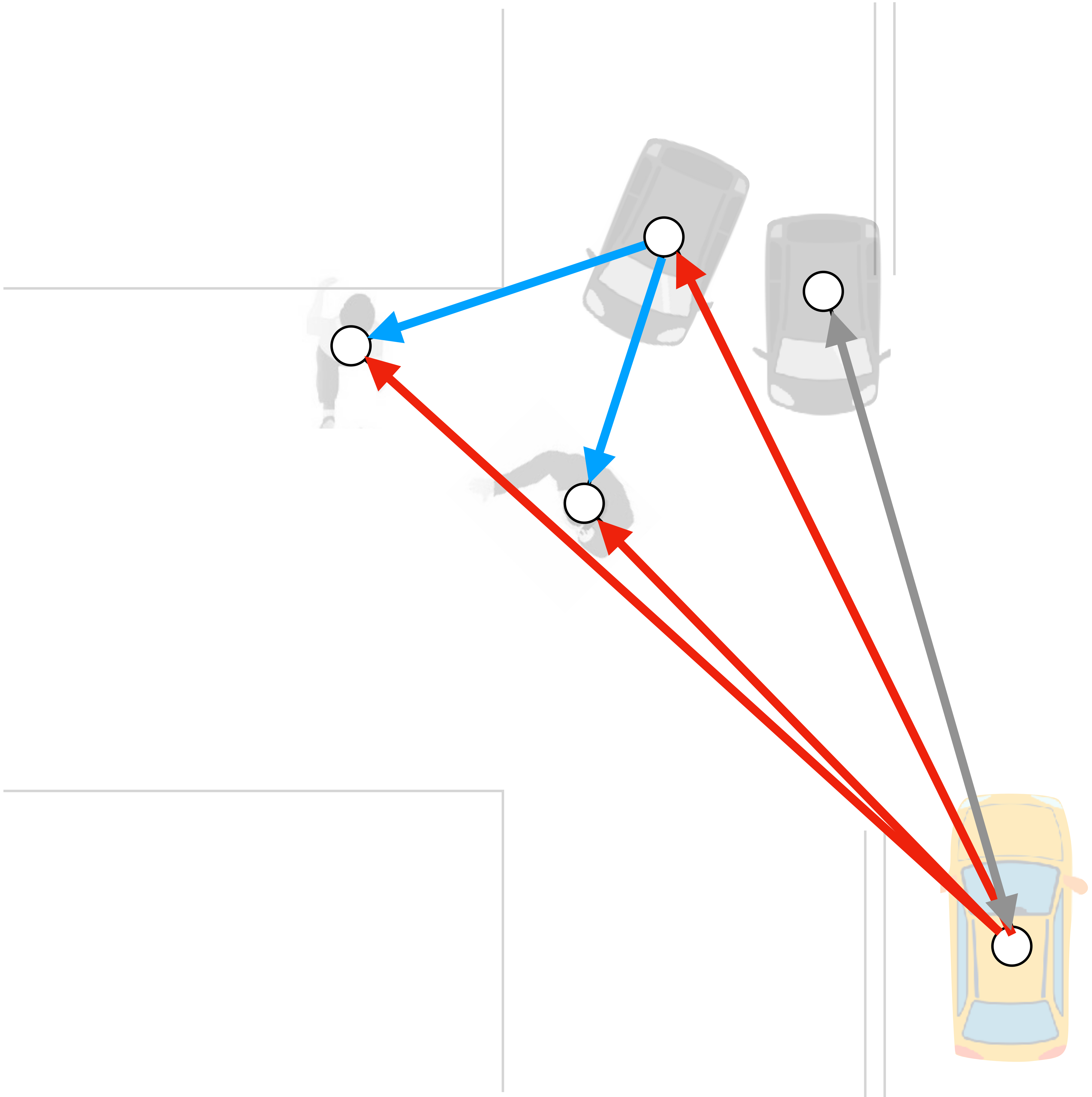
≡



Mode ≡ A single basin of forecast



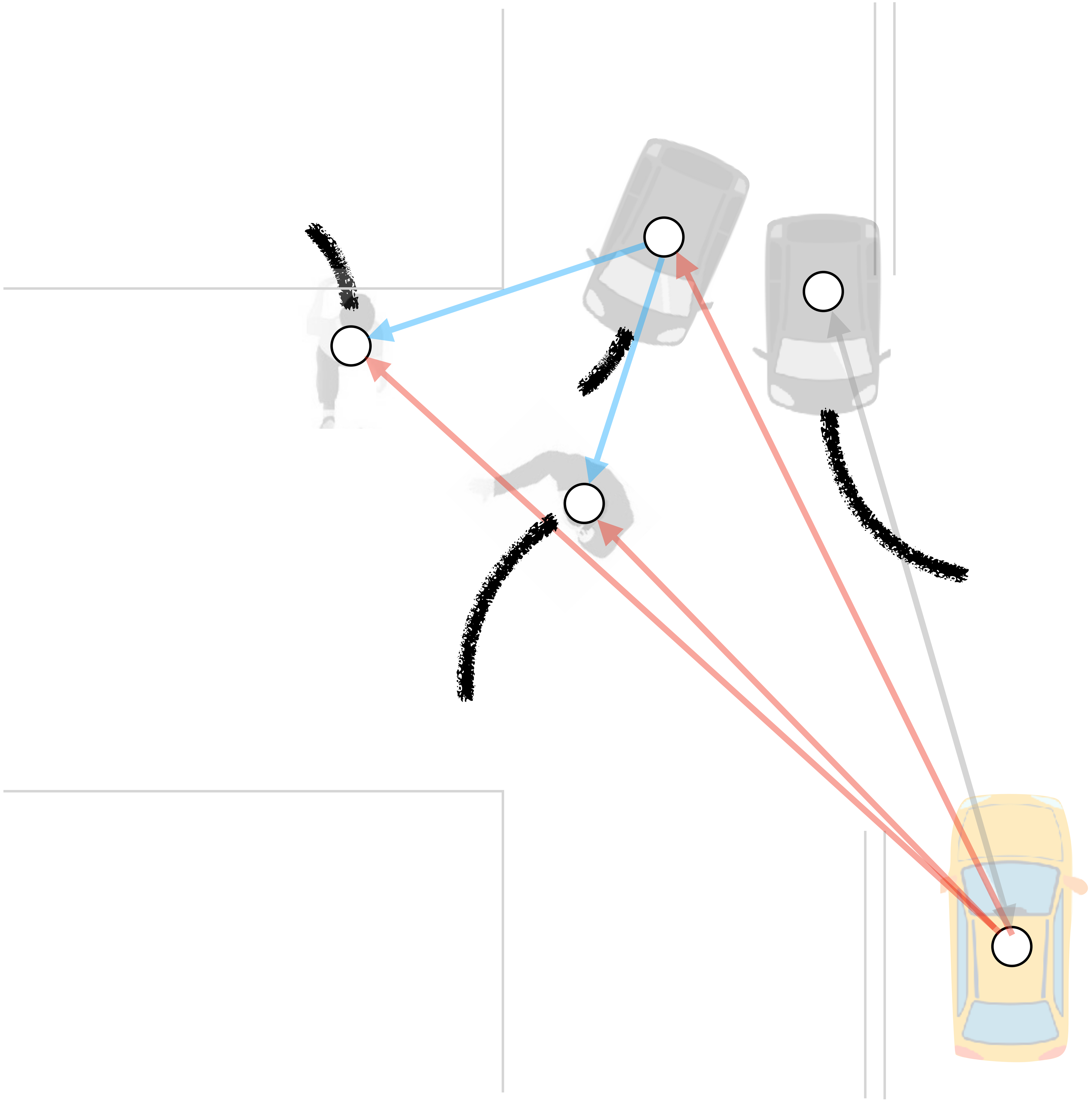
Condition on robot ~~plan~~ mode



Given a set of modes chosen by the robot

Infer what modes others are likely to choose

Message Passing on a Graph

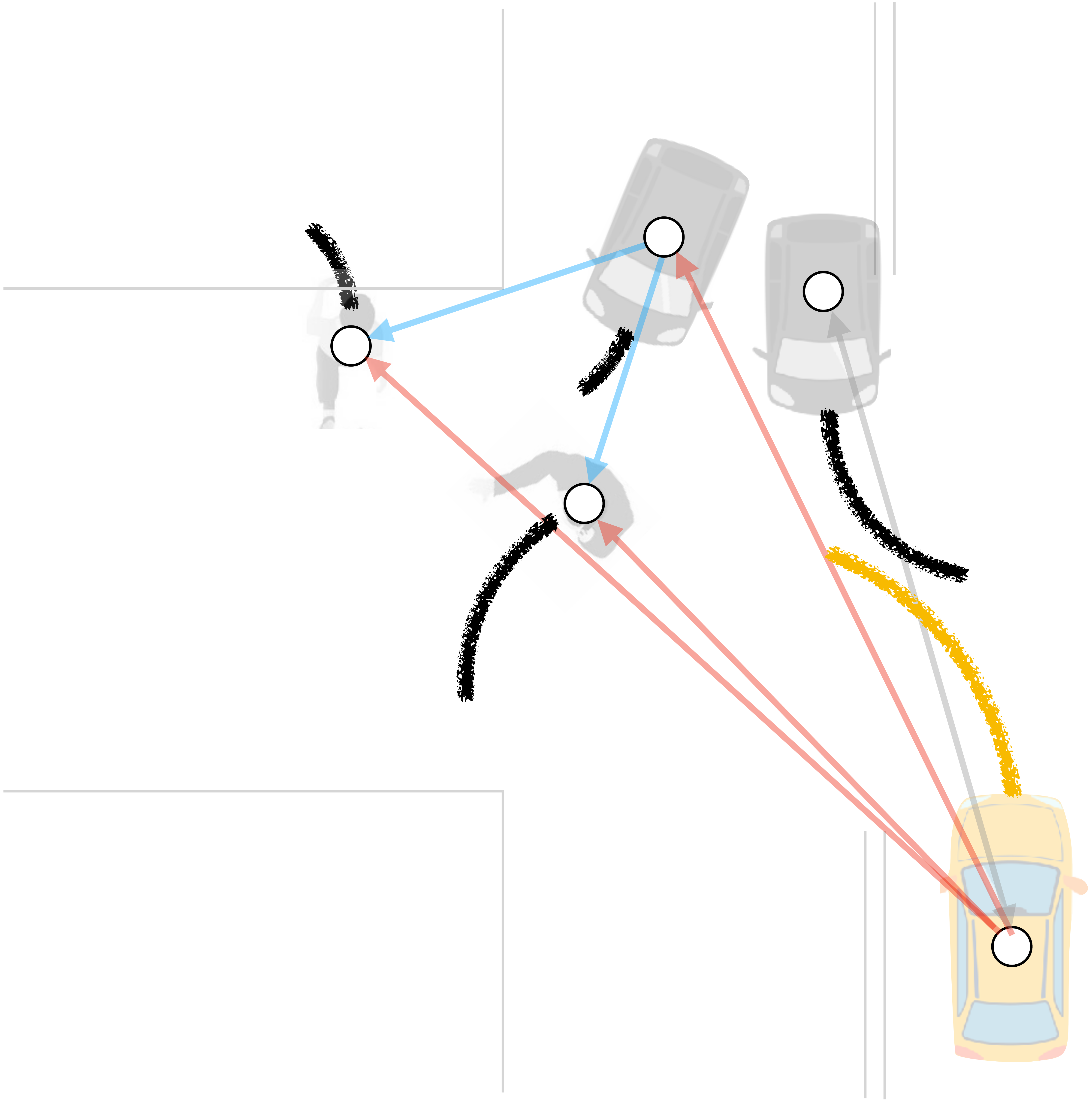


Given a set of modes chosen by the robot

Infer what modes others are likely to choose

Forecast actors given modes

Message Passing on a Graph

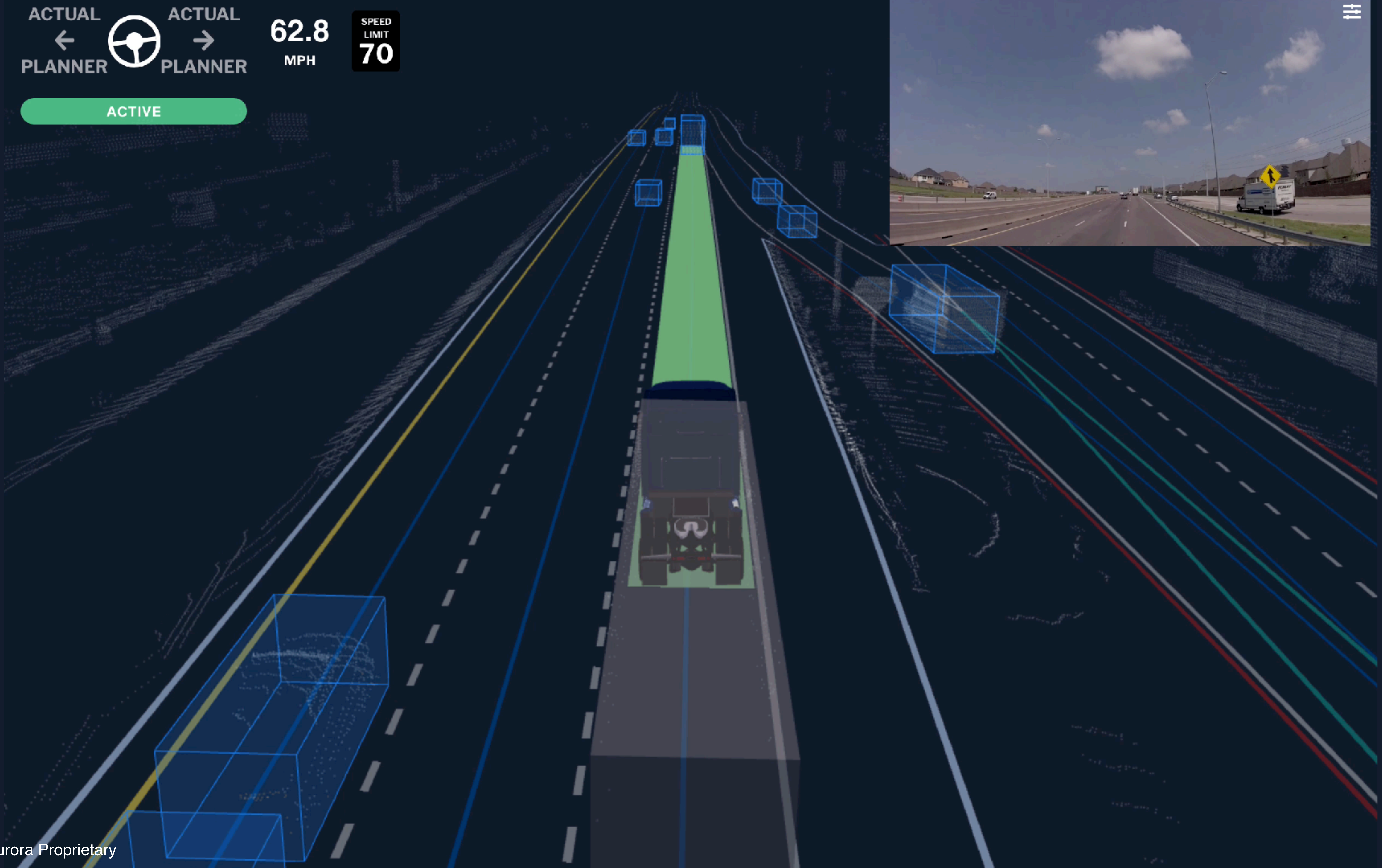


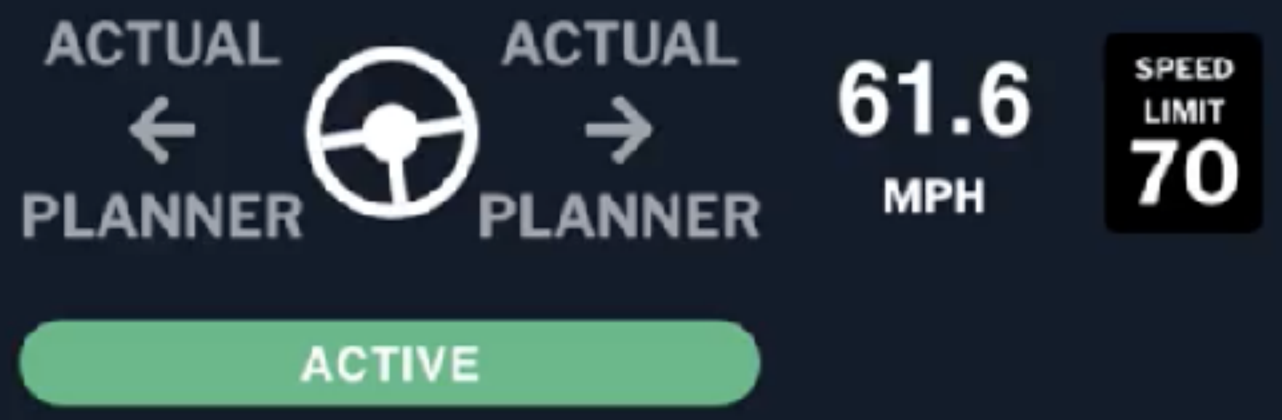
Given a set of modes chosen by the robot

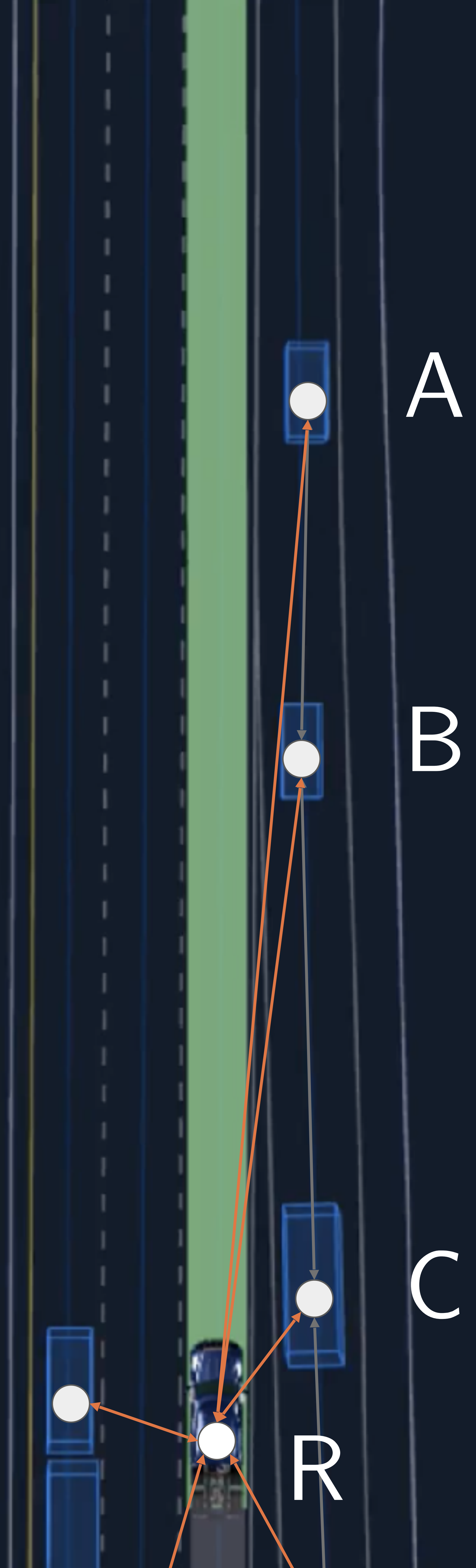
Infer what modes others are likely to choose

Forecast actors given modes

Plan given forecast


ACTUAL
PLANNER
ACTUAL
PLANNER
62.8
MPH
SPEED
LIMIT
70
ACTIVE

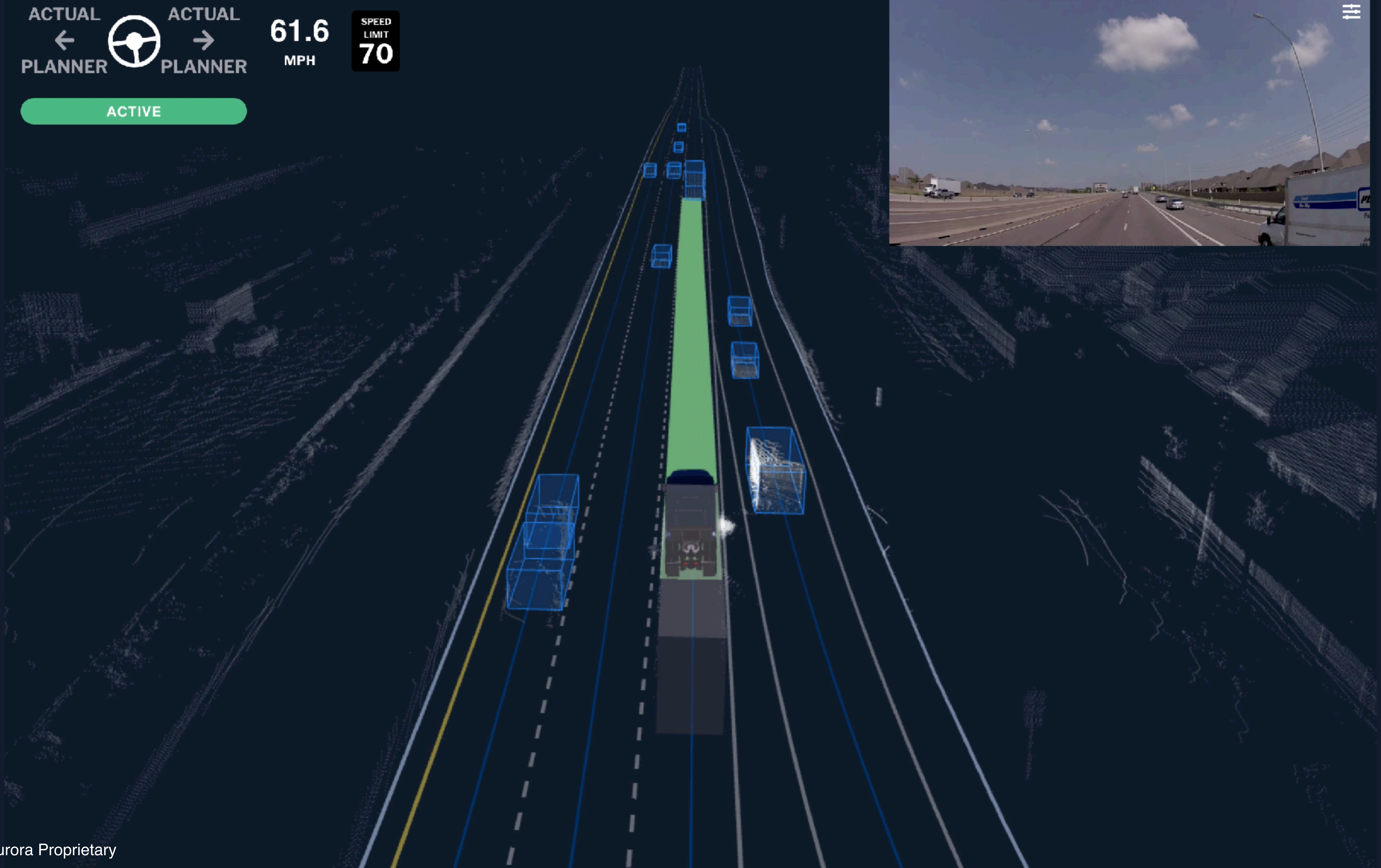




R Yields to A

R Yields to B

C Yields to R


ACTUAL
PLANNER
ACTUAL
PLANNER
61.6
MPH
SPEED
LIMIT
70
ACTIVE

Shaky foundations of forecasting

Are we using the right model?

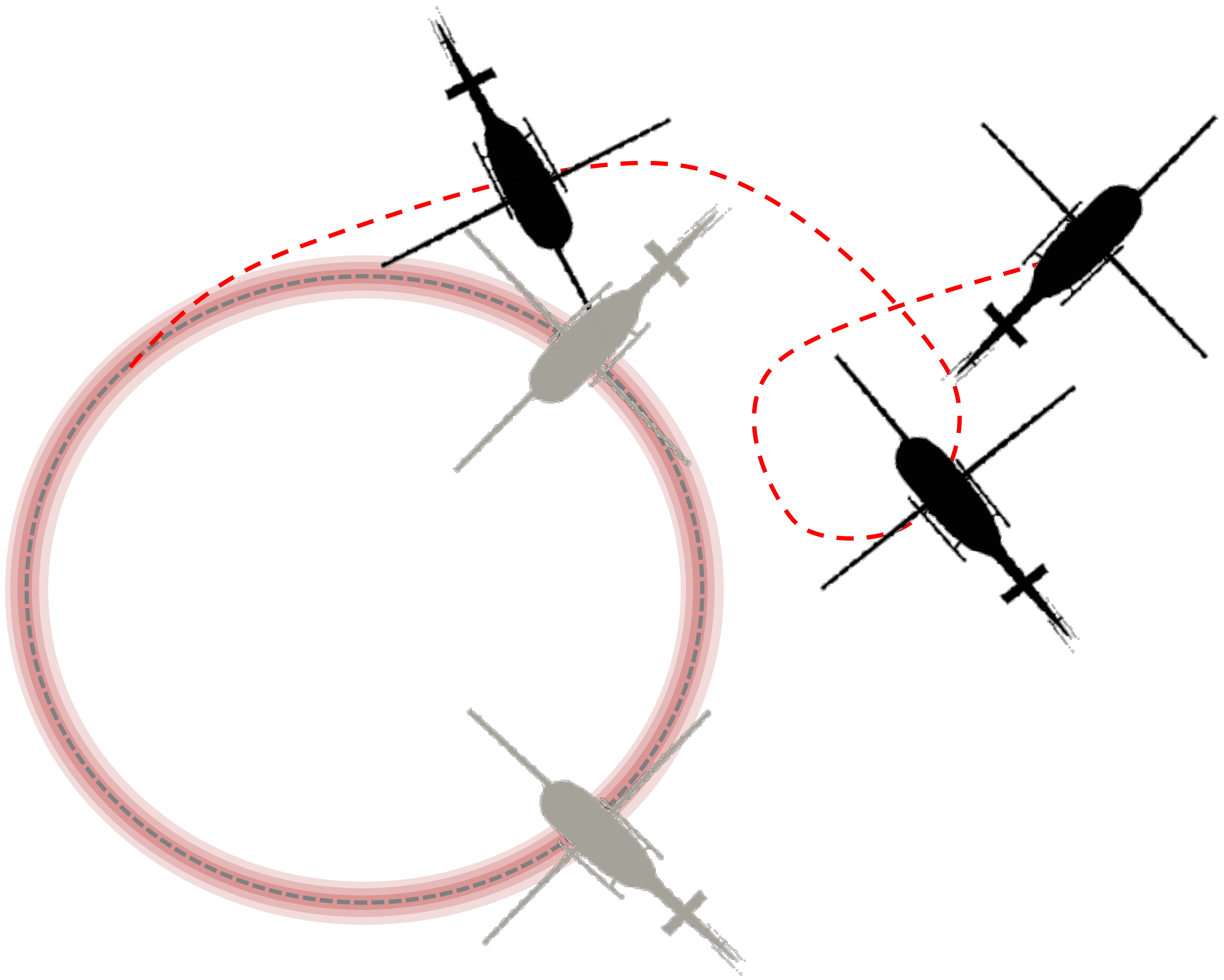
Conditional forecasting

Are we collecting data correctly?

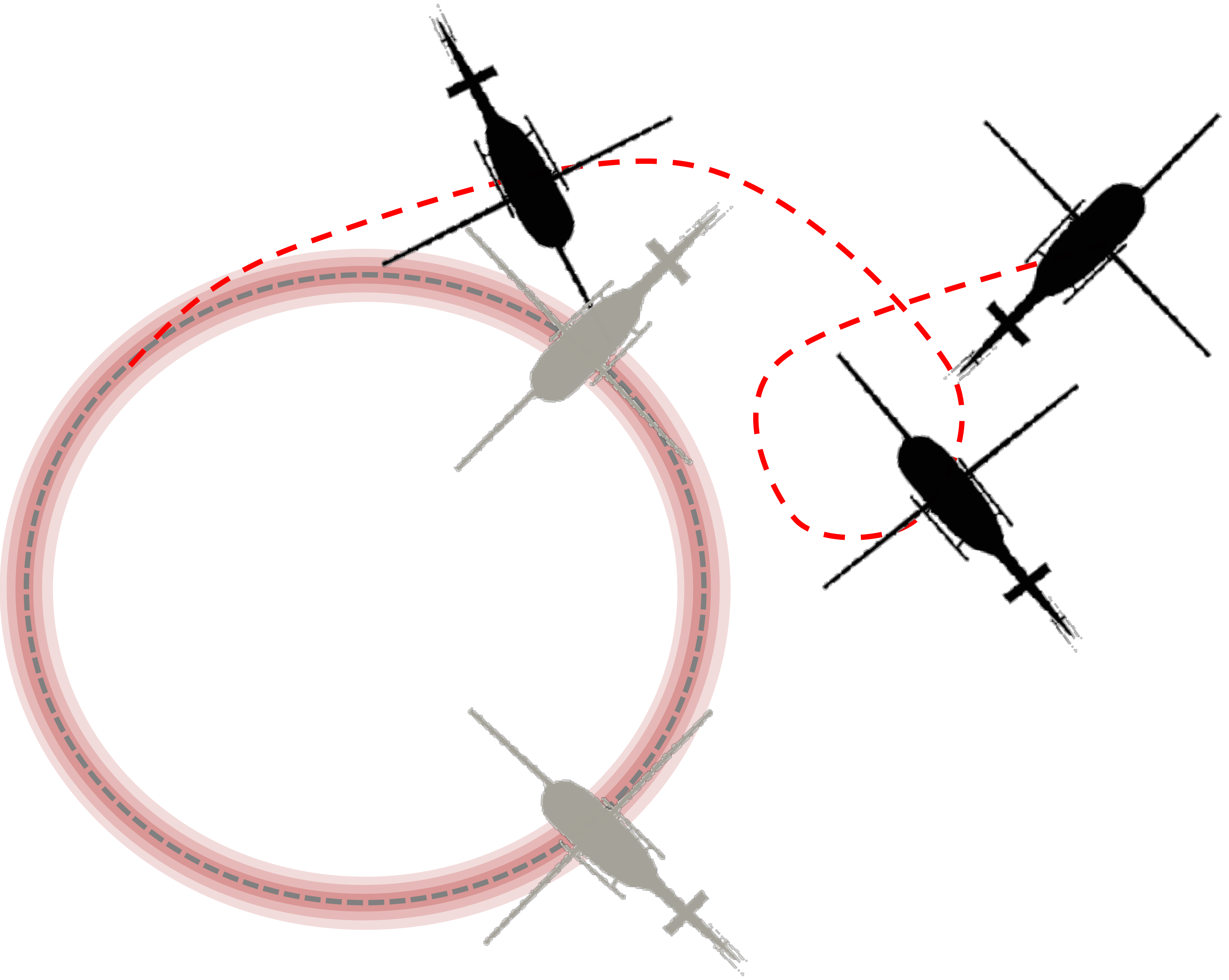
Are we using the right loss?

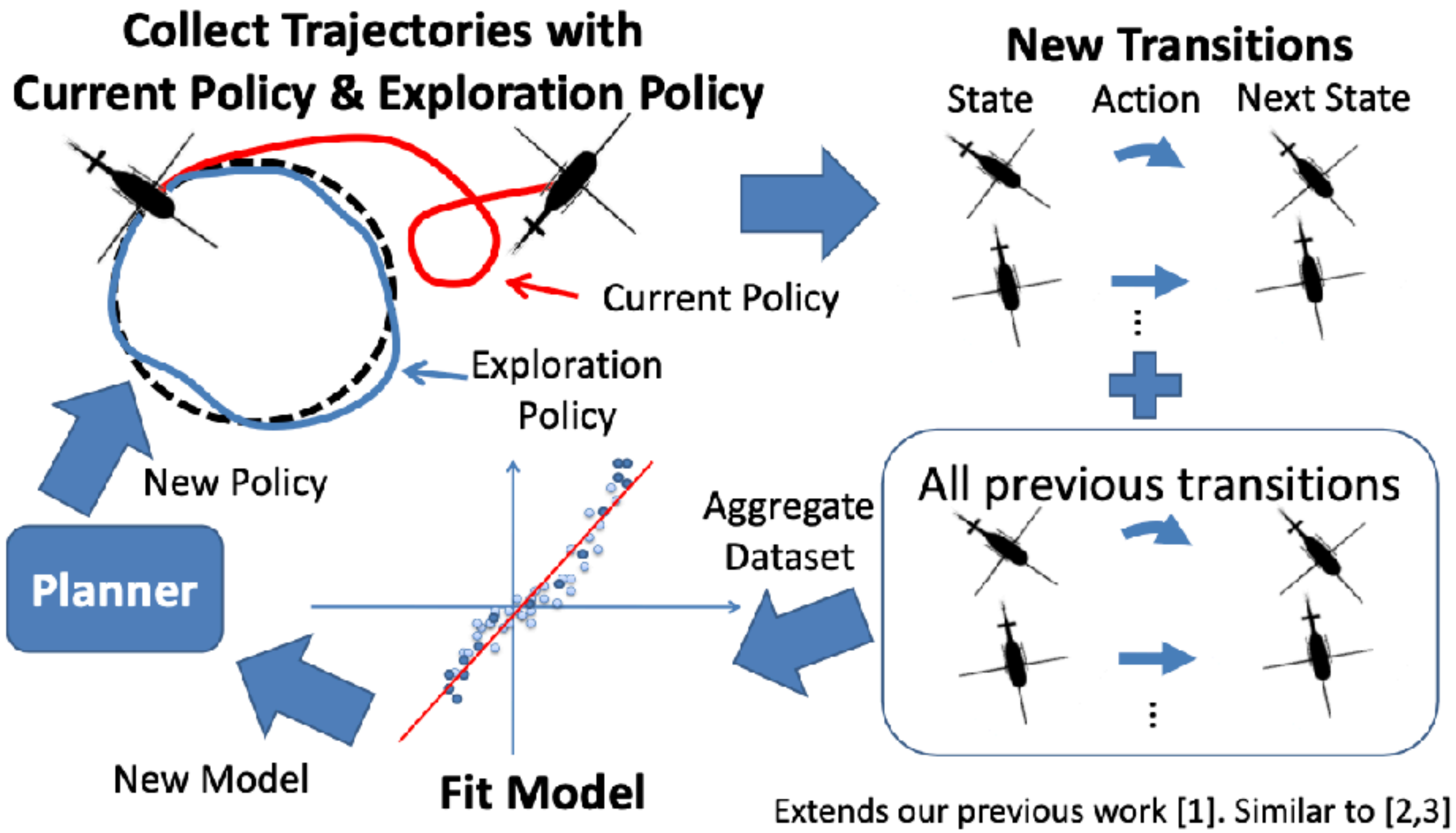
What happens when we deploy model?

We have seen this problem before!

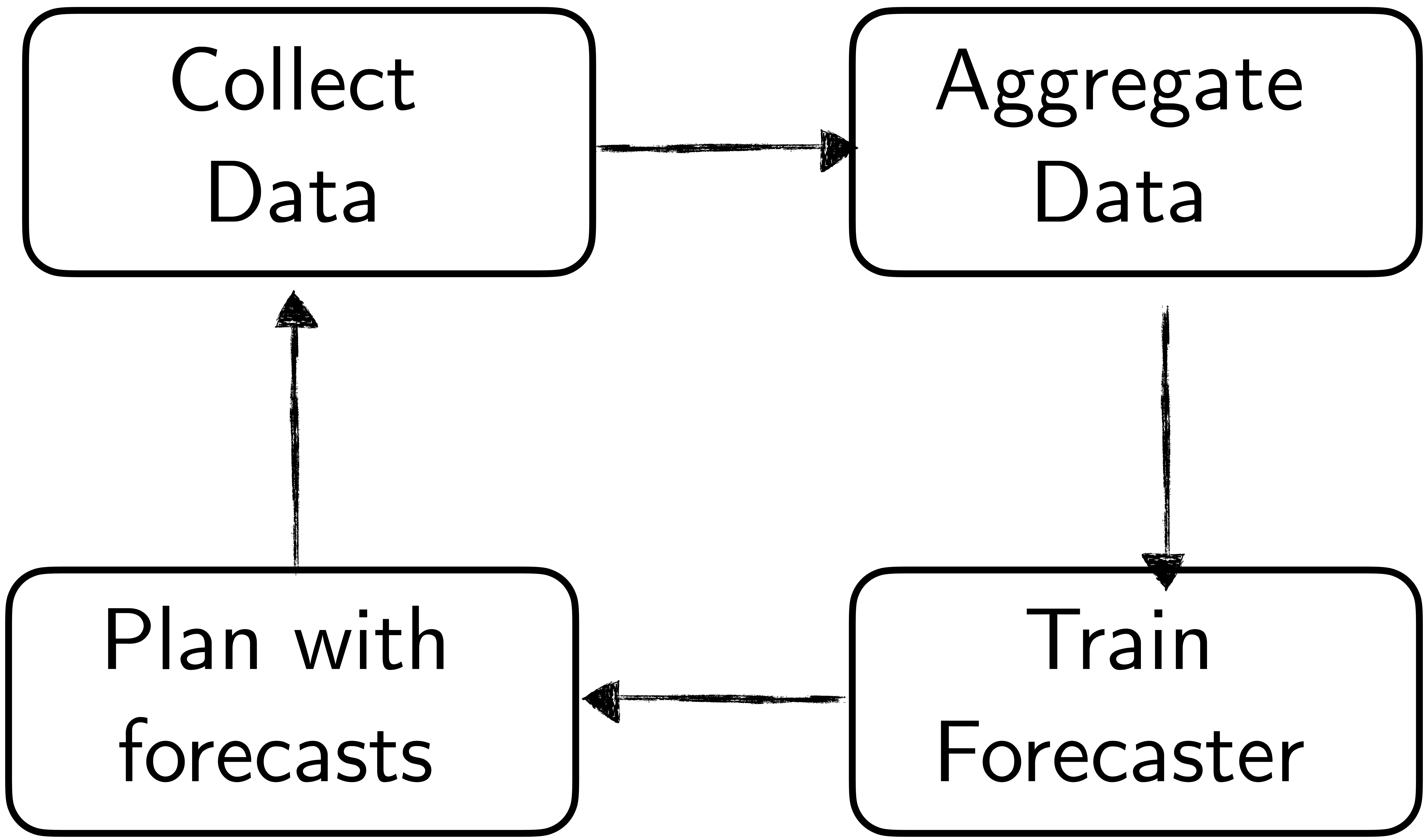


Solution: DAGGER for SysID





DAGGER for Forecasting!



Shaky foundations of forecasting

Are we using the right model?

Conditional forecasting

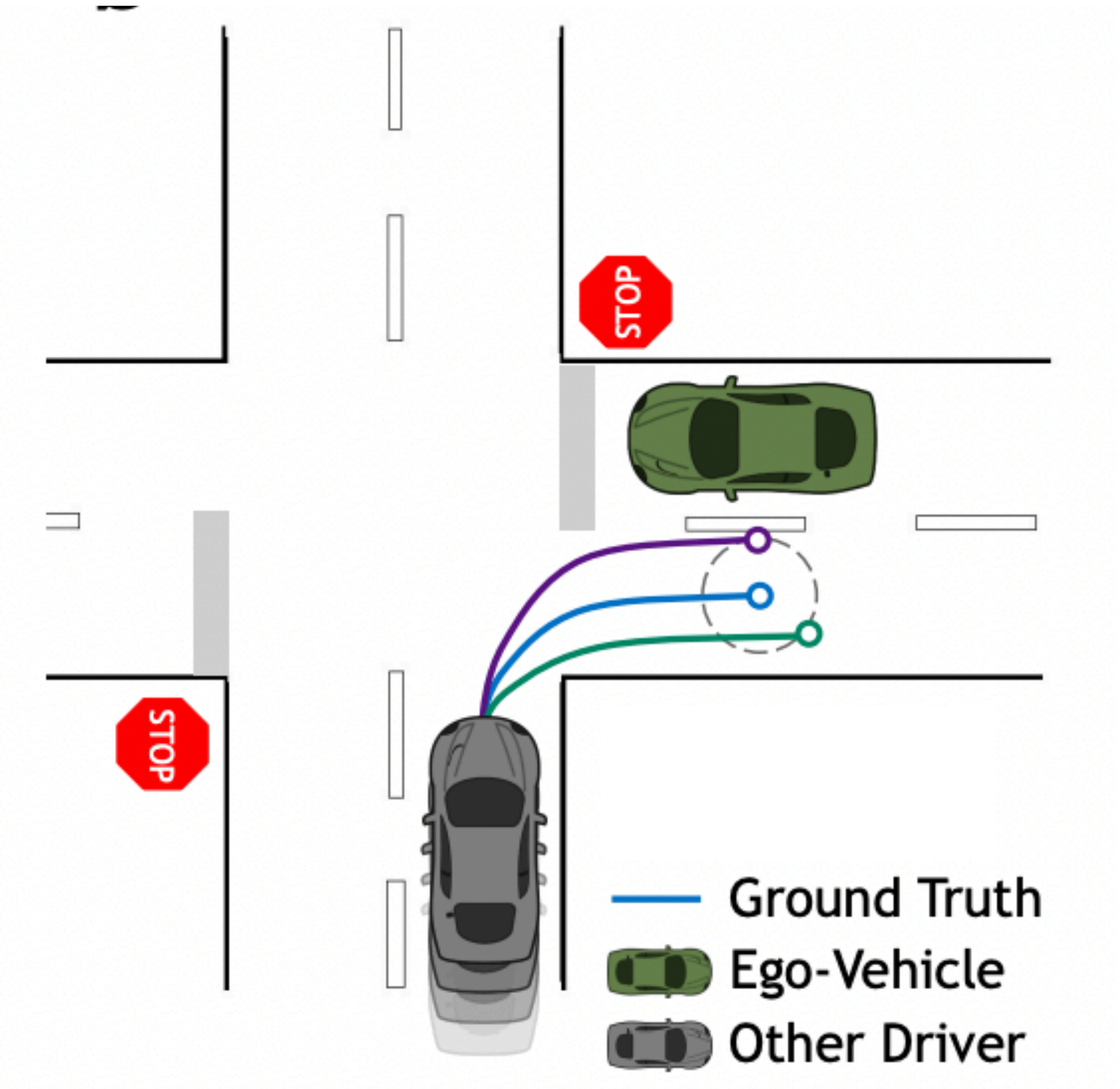
Are we collecting data correctly?

Interactively collect data

Are we using the right loss?

Is L2 loss the right loss function to use?

Is L2 loss the right loss function to use?



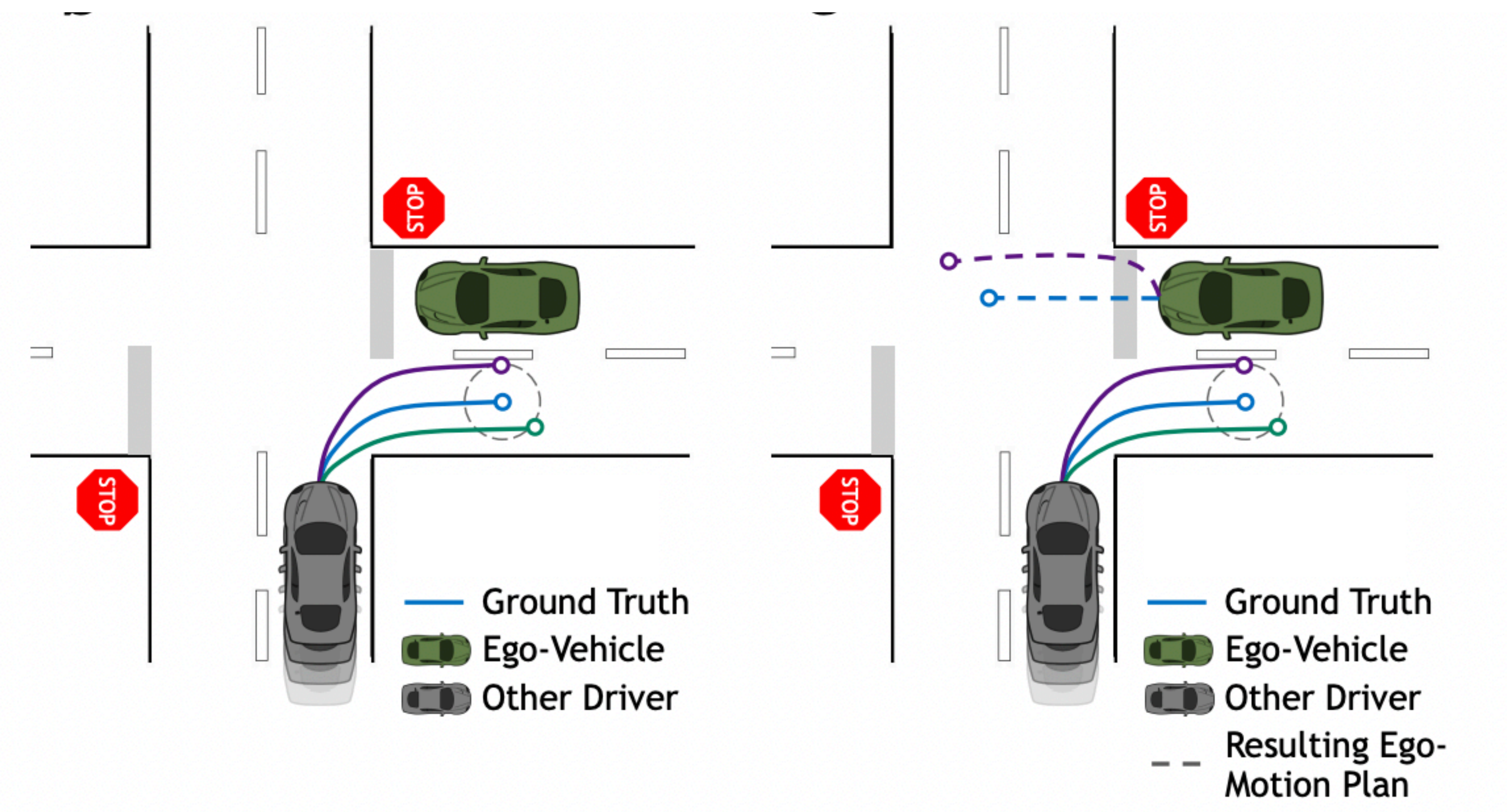
Rethinking Trajectory Forecasting Evaluation

Boris Ivanovic **Marco Pavone**
NVIDIA Research
{bivanovic, mpavone}@nvidia.com

Both forecast 1 and forecast 2 have the same L2 error

Which one would you prefer? Why?

What makes forecasts good?



Rethinking Trajectory Forecasting Evaluation

Boris Ivanovic **Marco Pavone**
NVIDIA Research
{bivanovic, mpavone}@nvidia.com

Forecasting is just a Model

Models are useful fictions

Forecasting <-> Model-based RL

Conditional Forecasts

Model

$$P(s_{t:t+k} \,|\, s_{t:t-k}, \xi_{plan})$$

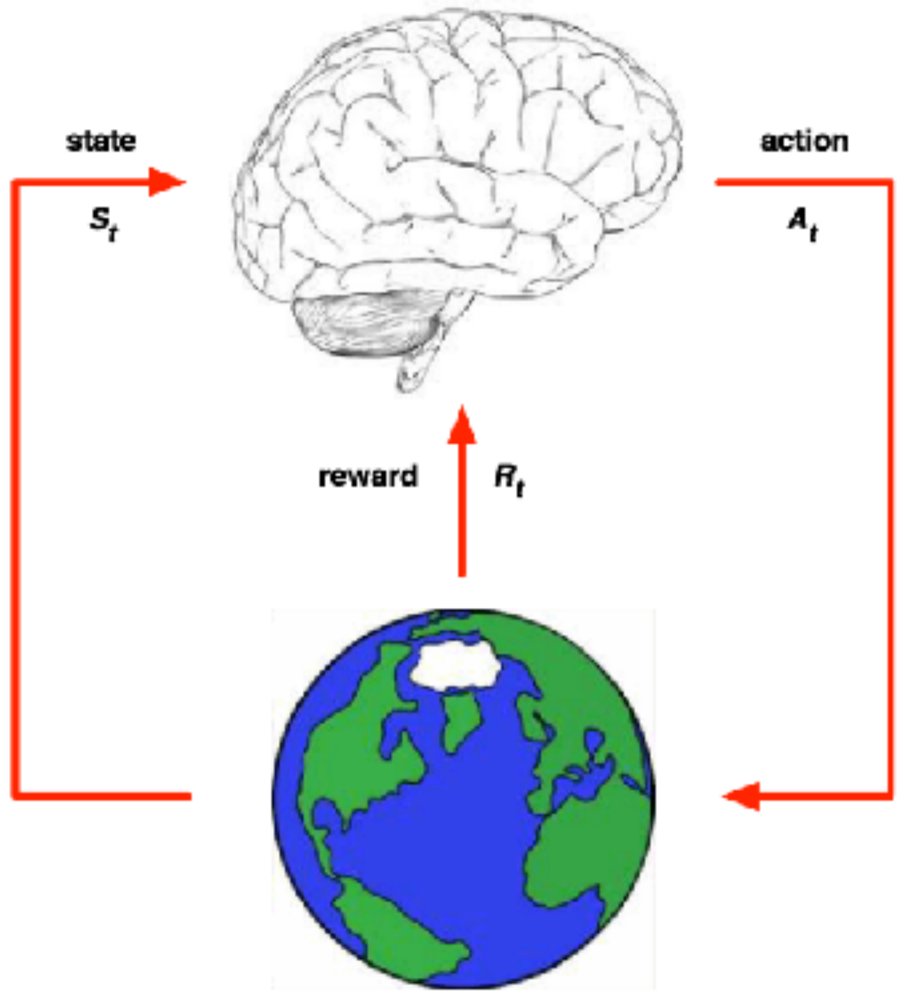
$$M(s_{t+1} \,|\, s_t, a_t)$$

We know how to solve model-based RL (previous lecture!)

What is the ONE true loss that we care about?

Performance Difference between our policy and the expert

$$J_{M^*}(\hat{\pi}) - J_{M^*}(\pi^*)$$



Recall: Perf Diff implies matching Values

$$J_{M^*}(\pi^*) - J_{M^*}(\hat{\pi})$$

$$= \mathbb{E}_{s^* \sim \pi^*}\left[A^{\hat{\pi}}(s^*, a^*)\right] \quad + T\mathbb{E}_{s,a \sim \pi^*}\left[E_{s' \sim \hat{M}} V^{\hat{\pi}}(s') - E_{s'' \sim M^*} V^{\hat{\pi}}(s'')\right]$$

Advantage of expert in model

Value matching on expert states

$$+ T\mathbb{E}_{s,a \sim \hat{\pi}}\left[E_{s' \sim \hat{M}} V^{\hat{\pi}}(s') - E_{s'' \sim M^*} V^{\hat{\pi}}(s'')\right]$$

Value matching on learner states

A simple loss function

Replace L2 Loss

$$||\xi - \xi_{gt}||^2$$

With cost difference loss

$$||c(\xi) - c(\xi_{gt})||^2$$

Where $c(\,.\,)$ are a set of cost features (proximity, jerkiness etc)

Shaky foundations of forecasting

Are we using the right model?

Conditional forecasting

Are we collecting data correctly?

Interactively collect data

Are we using the right loss?

Performance Difference

Do these ideas extend beyond self-driving?

K. Kedia, P. Dan, A. Bhardwaj, S. Choudhury ManiCast: Collaborative Manipulation with Cost-Aware Human Forecasting. *CORL 2023.*

K. Kedia, P. Dan, A. Bhardwaj, S. Choudhury INTERACT: Transformer Models for Human Intent Prediction Conditioned on Robot Actions. *ICRA 2024. (in submission)*

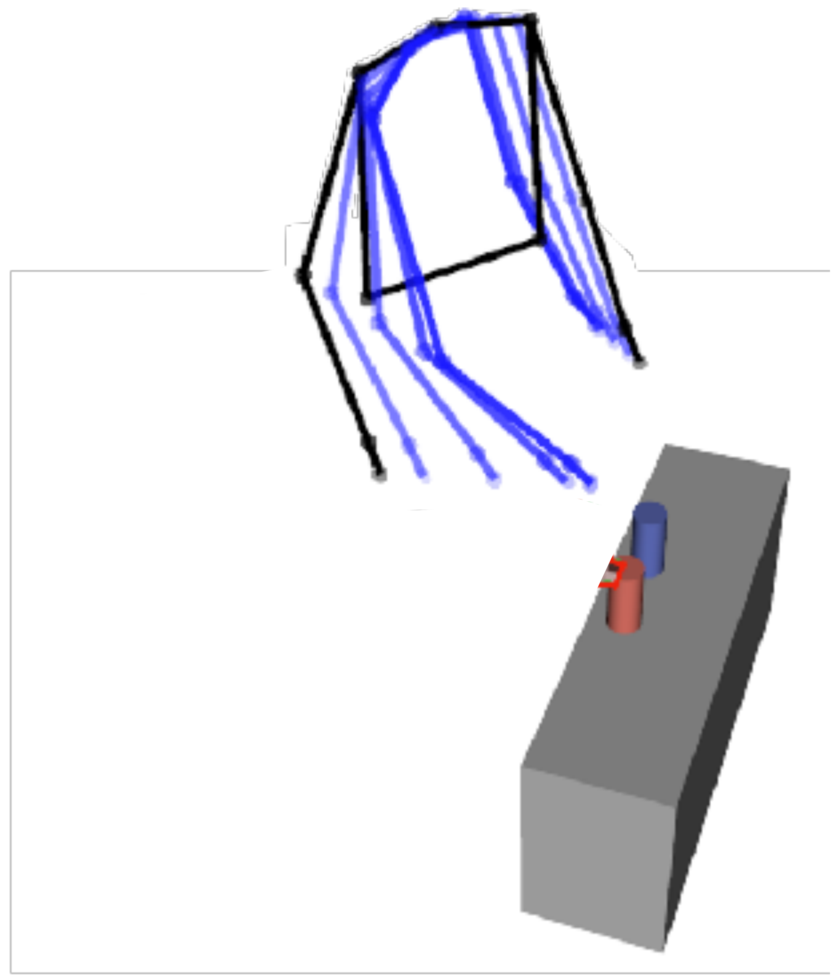
K. Kedia, P. Dan, A. Bhardwaj, S. Choudhury. A Game-Theoretic Framework for Joint Forecasting and Planning. *IROS 2023.*

Goal: Predict human motion *conditioned* on robot goal

Goal: Predict human motion *conditioned* on robot goal

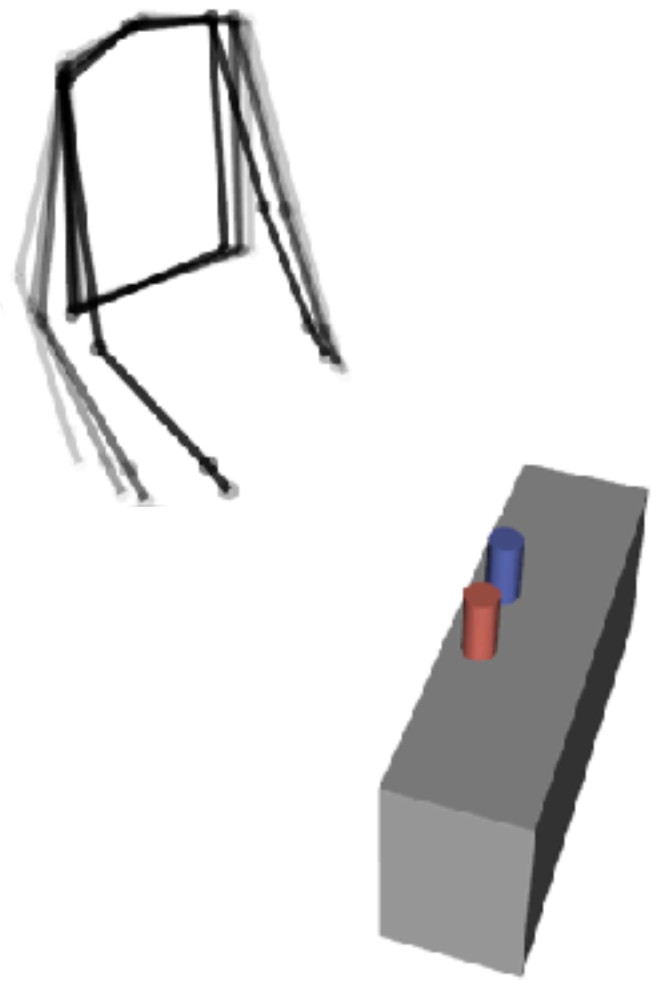
$$P(\quad \mid \quad)$$

Human future

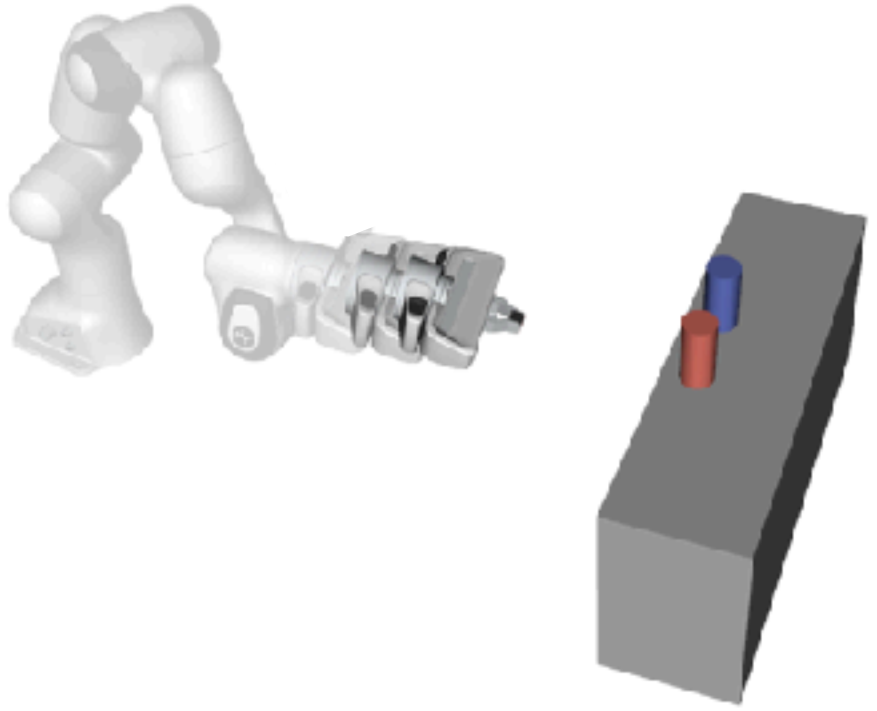


Context

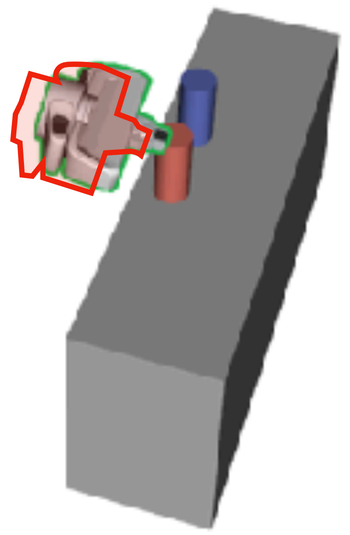
Human history



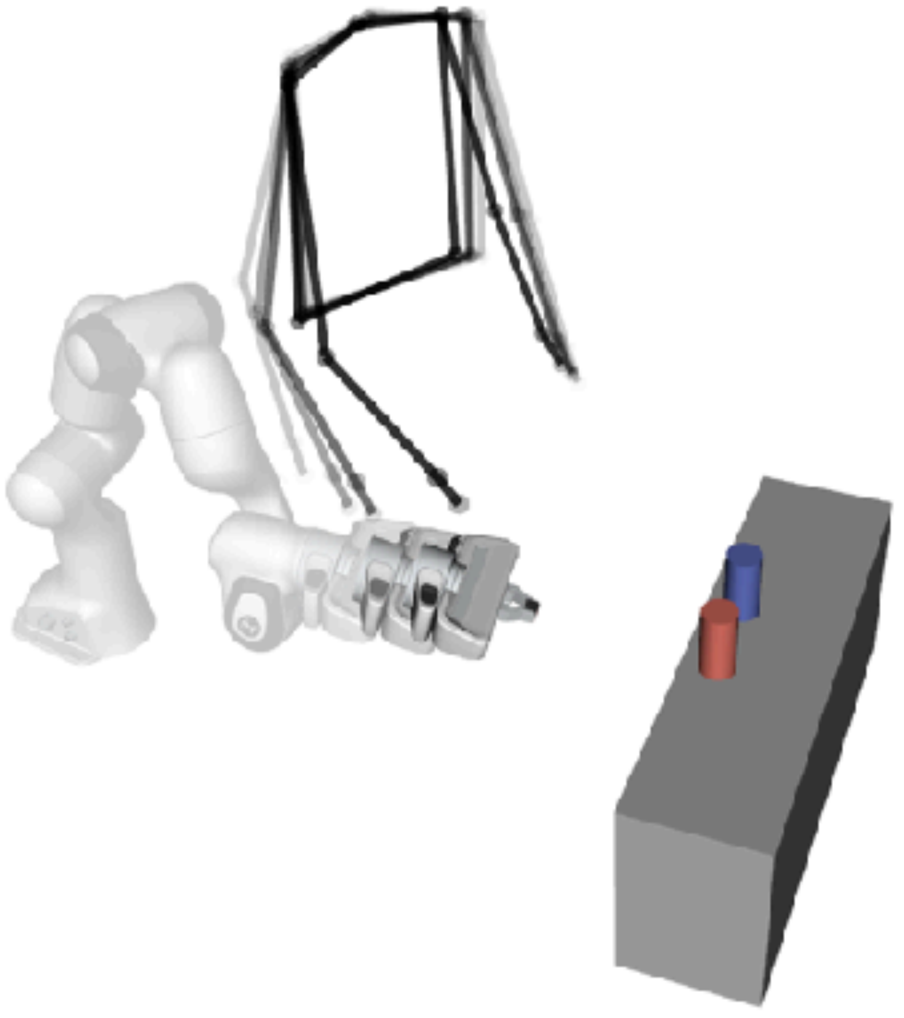
Robot history



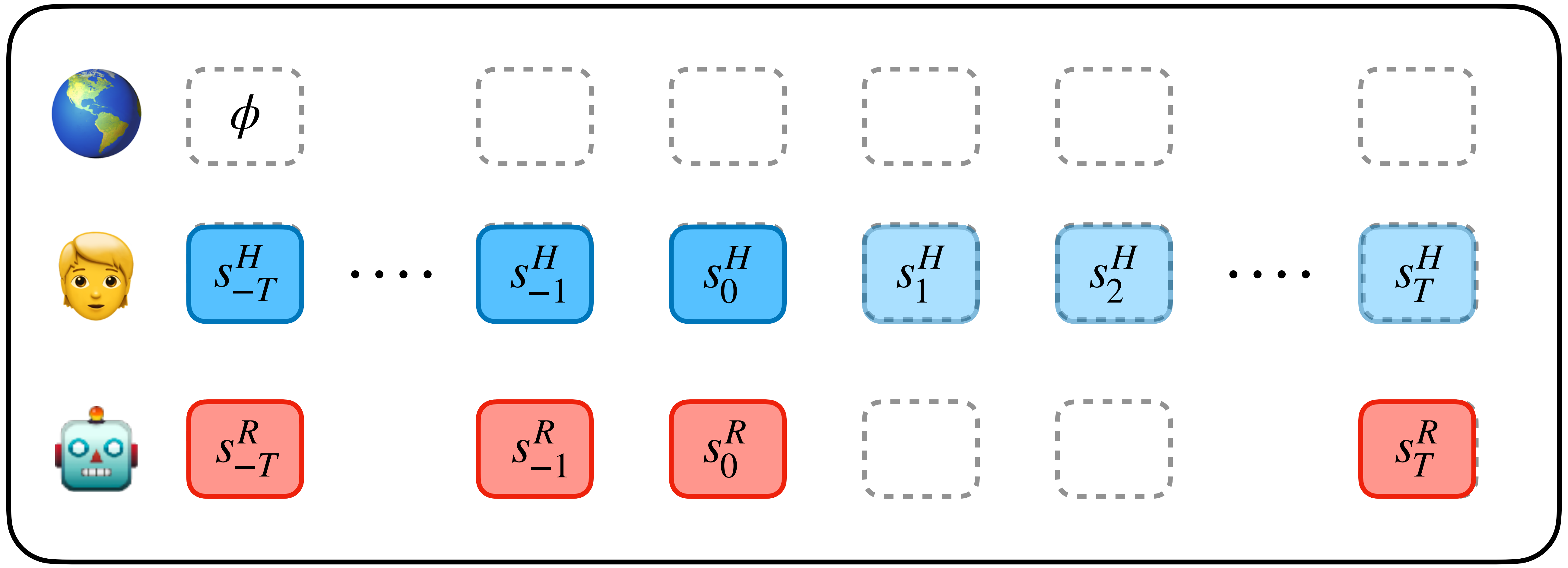
Robot goal



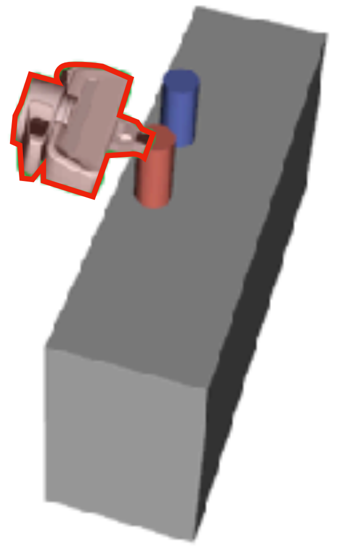
Framework: Sequence prediction



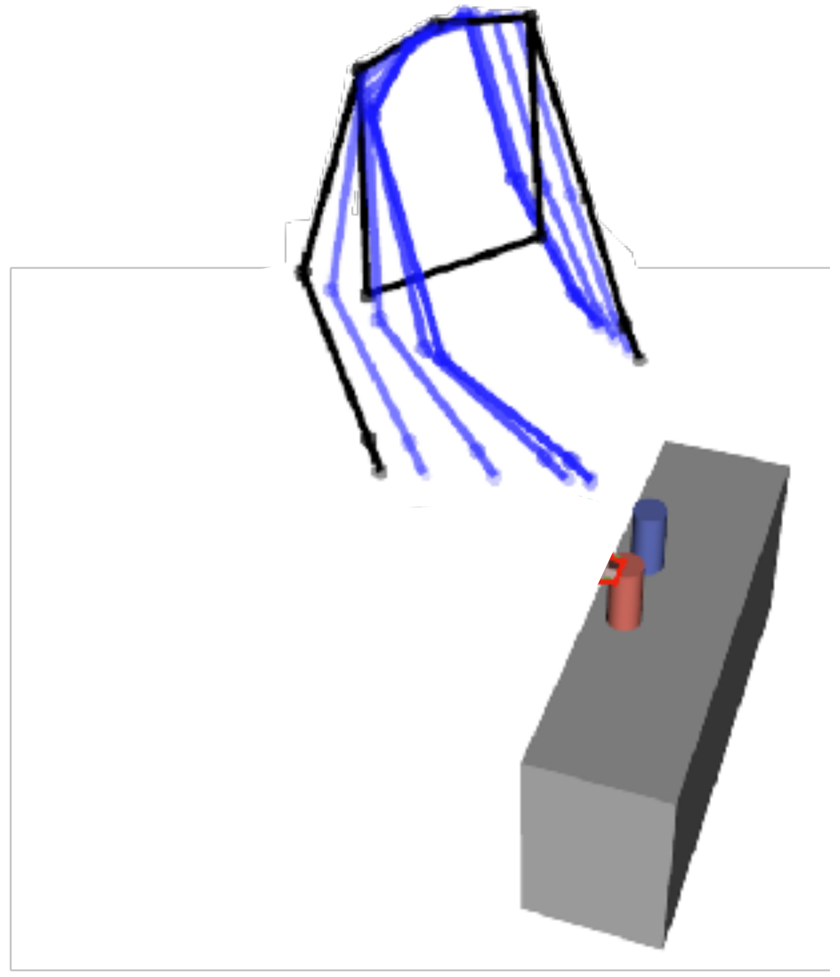
Human history
Robot history



Transformer with
masked attention

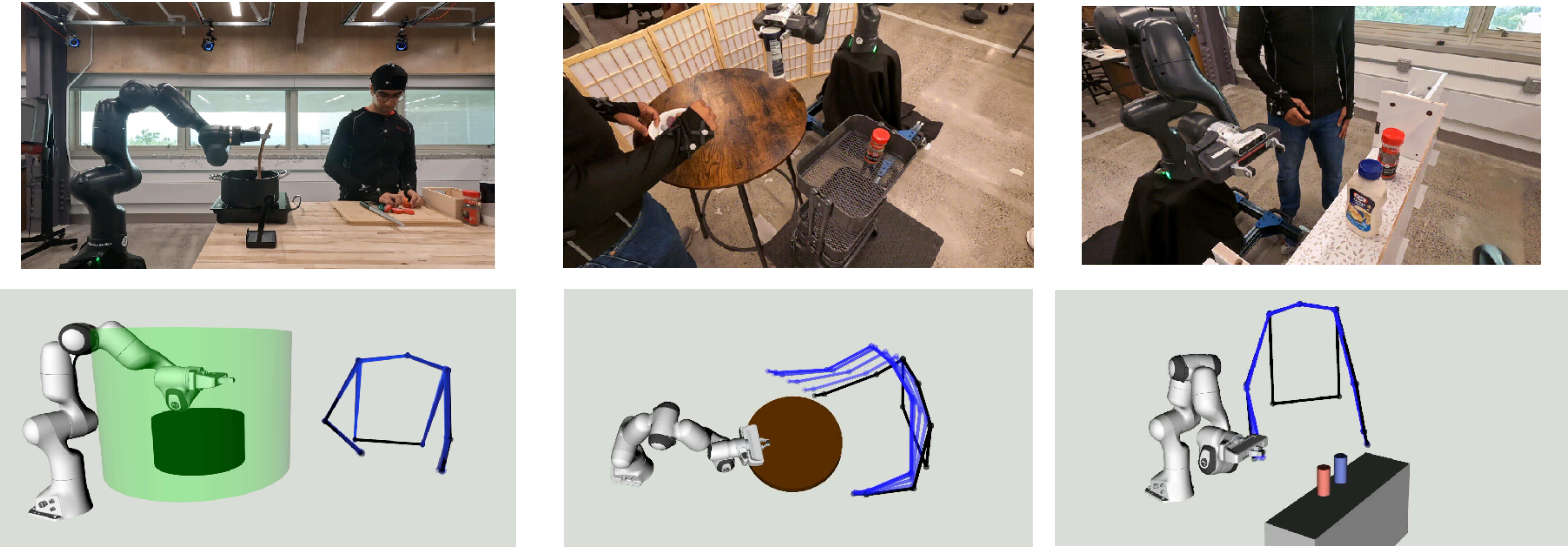


Robot
Goal



Human future

Results: Evaluate across different users and tasks

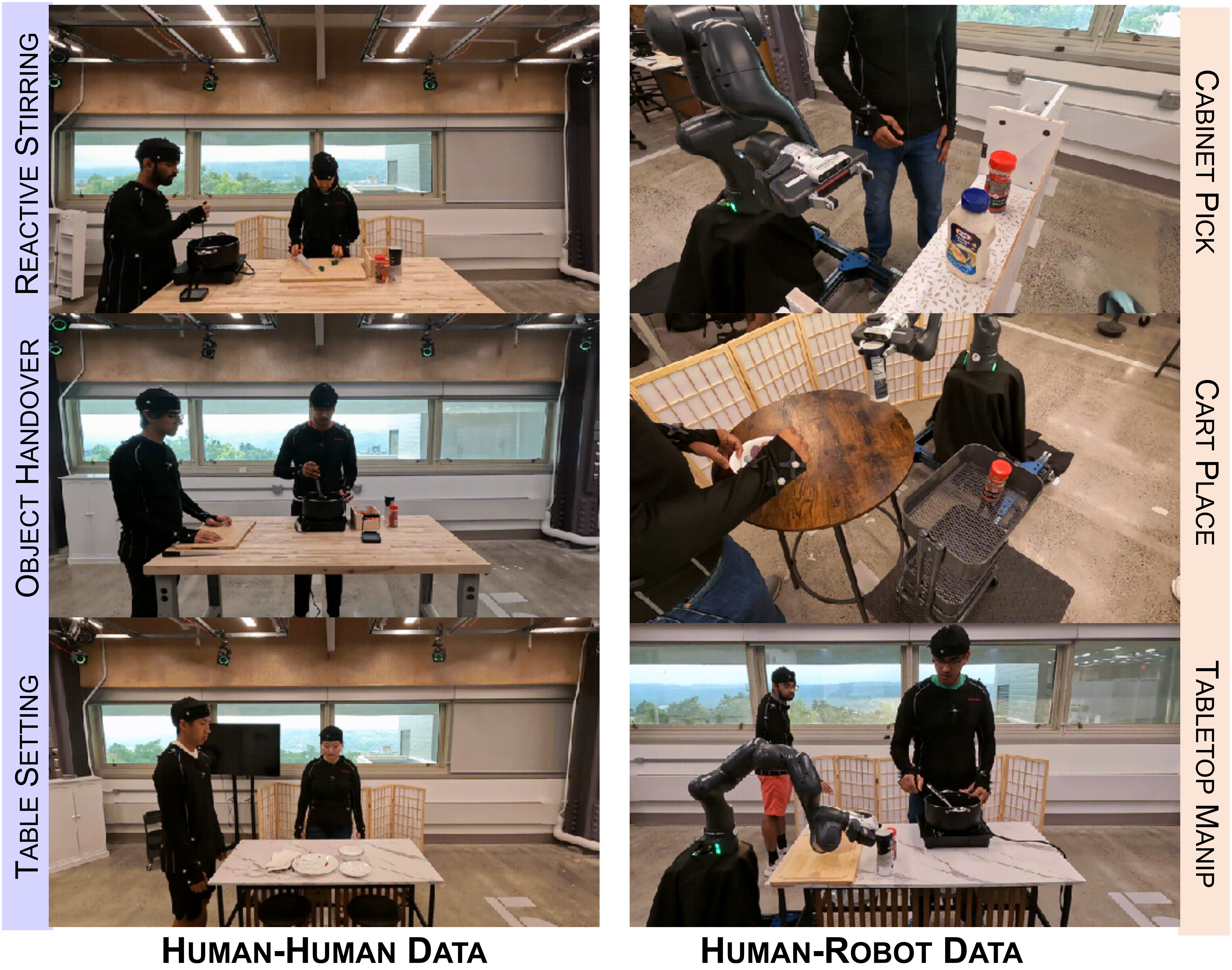


New Dataset: Collaborative Manipulation Dataset (CoMaD)

Over 4 hours of human motion

3 different home tasks

270 episodes of human-human interaction

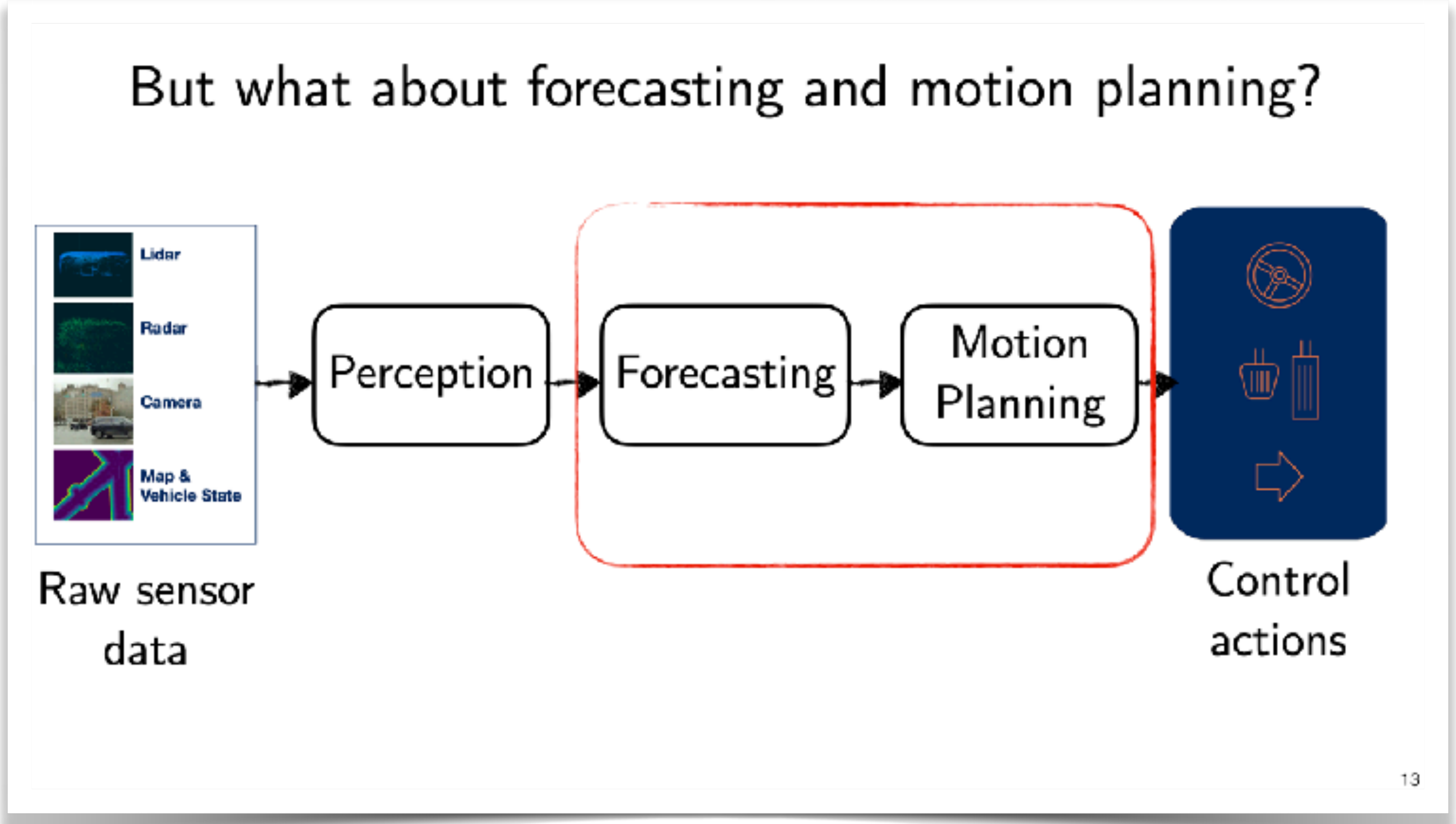


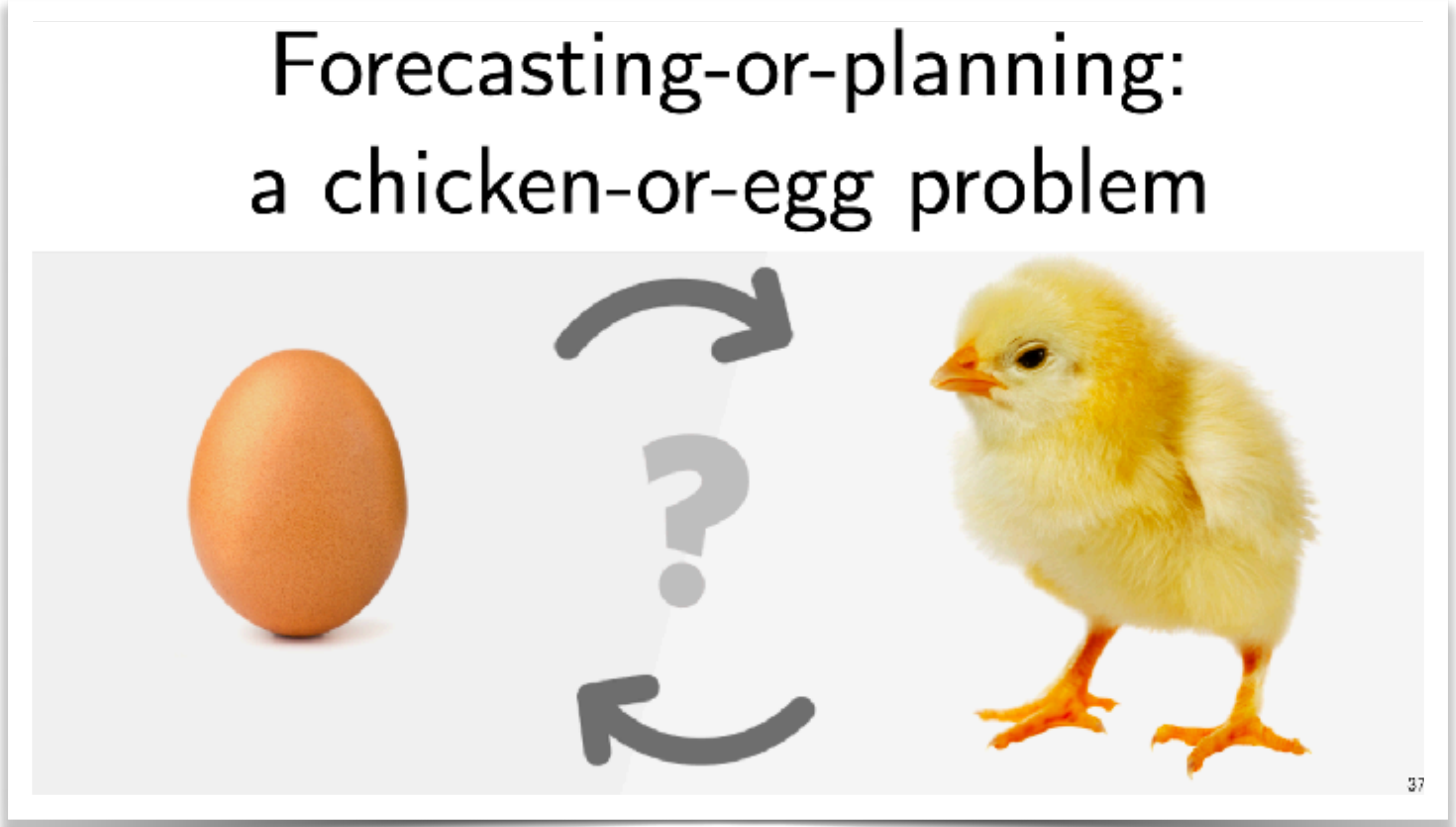
30 minutes of paired human-robot Interaction

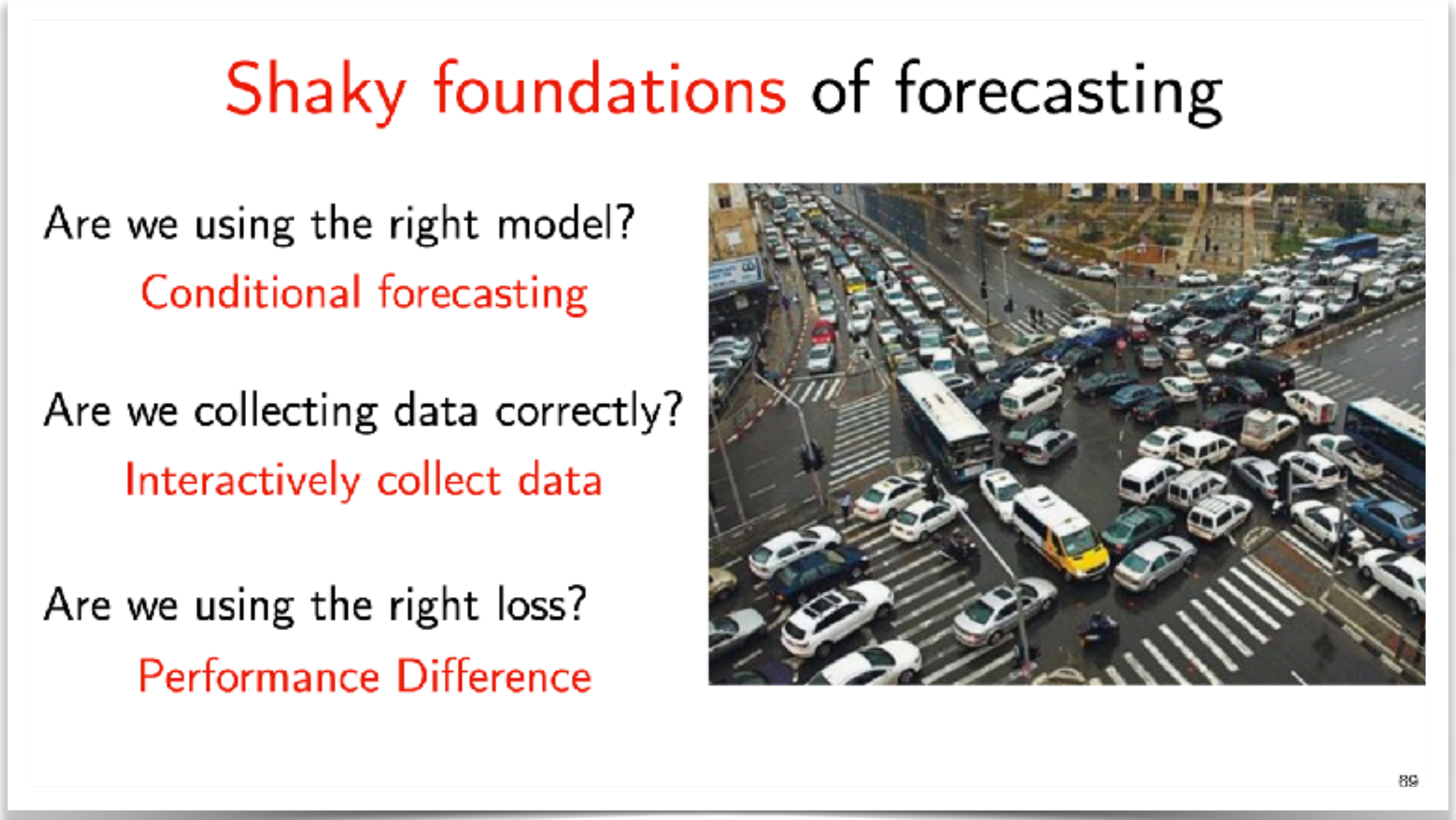
3 different home tasks

217 episodes of human-robot interaction

tl;dr


But what about forecasting and motion planning?
Lidar
Radar
Camera
Map & Vehicle State
Raw sensor data
Perception
Forecasting
Motion Planning
Control actions
13


Forecasting-or-planning:
a chicken-or-egg problem
37


Shaky foundations of forecasting
Are we using the right model?
Conditional forecasting
Are we collecting data correctly?
Interactively collect data
Are we using the right loss?
Performance Difference