PRECOG: PREdiction Conditioned On Goals in Visual Multi-Agent Settings

Ethan Pronovost & Ram Kodey

Intro

Motion forecasting for planning

- Joint motion forecasting
  - Model all agents together, can model interactions
  - Probability distribution over joint trajectories
- Ego-conditioned forecasting
  - Manually specify robot's behavior, sample for other agents
- Goal-conditioned planning
  - Optimize robot behavior based on a goal and model likelihood

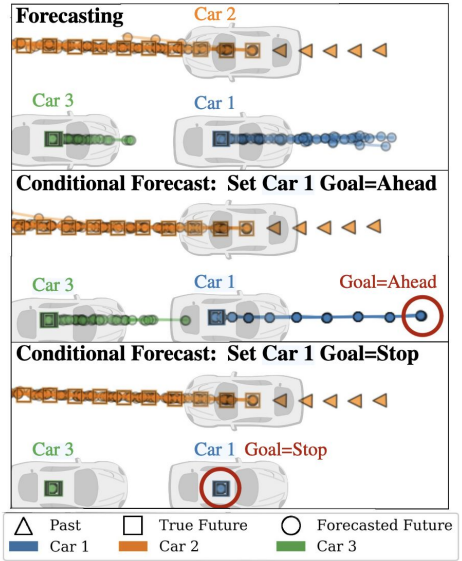


Figure 2: Conditioning the model on different Car 1 goals produces different predictions: here it forecasts Car 3 to move if Car 1 yields space, or stay stopped if Car 1 stays stopped.

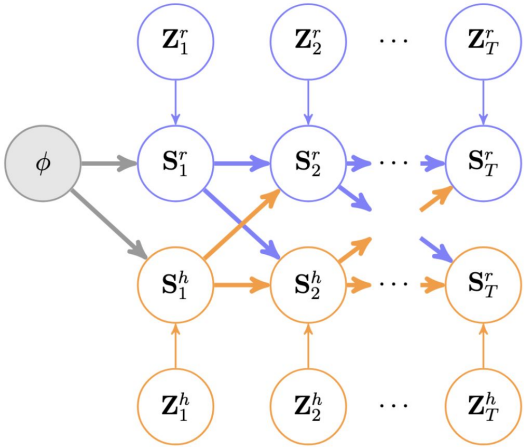
ArXiv
Website

 Academic researcher

Joint rollouts

- State of agent *a* at time *t* depends on previous states of all agents
- Model learns: $p(\mathbf{S}_t^a \mid \mathbf{S}_{1:t-1}^{1:A}, \phi)$
- Full probability model: $p(\mathbf{S} \mid \phi) = \prod_{t=1}^{T} \prod_{a=1}^{A} p(\mathbf{S}_t^a \mid \mathbf{S}_{1:t-1}^{1:A}, \phi)$

ϕ Placeholder for conditional data

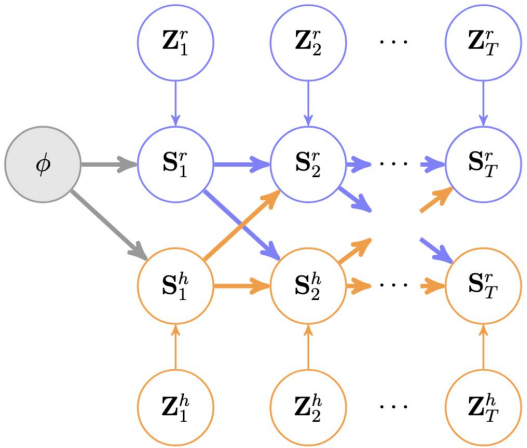


Joint rollouts

- State of agent *a* at time *t* depends on previous states of all agents
- Model learns: $p(\mathbf{S}_t^a \mid \mathbf{S}_{1:t-1}^{1:A}, \phi)$
- Full probability model: $p(\mathbf{S} \mid \phi) = \prod_{t=1}^{T} \prod_{a=1}^{A} p(\mathbf{S}_t^a \mid \mathbf{S}_{1:t-1}^{1:A}, \phi)$

ϕ Placeholder for conditional data

$$p(\mathbf{S}_t^a \mid \mathbf{S}_{1:t-1}^{1:A}, \phi) = \mathcal{N}(\mathbf{S}_t^a; \mu_\theta^a(\mathbf{S}_{1:t-1}^{1:A}, \phi), \Sigma_\theta^a(\mathbf{S}_{1:t-1}^{1:A}, \phi))$$



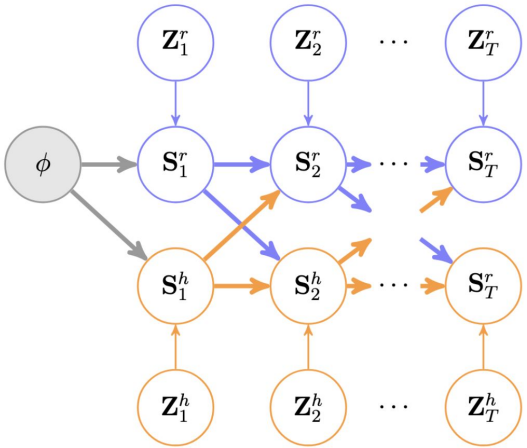
Joint rollouts

- State of agent *a* at time *t* depends on previous states of all agents
- Model learns: $p(\mathbf{S}_t^a \mid \mathbf{S}_{1:t-1}^{1:A}, \phi)$
- Full probability model: $p(\mathbf{S} \mid \phi) = \prod_{t=1}^{T} \prod_{a=1}^{A} p(\mathbf{S}_t^a \mid \mathbf{S}_{1:t-1}^{1:A}, \phi)$

ϕ Placeholder for conditional data

$$p(\mathbf{S}_t^a \mid \mathbf{S}_{1:t-1}^{1:A}, \phi) = \mathcal{N}(\mathbf{S}_t^a; \mu_\theta^a(\mathbf{S}_{1:t-1}^{1:A}, \phi), \Sigma_\theta^a(\mathbf{S}_{1:t-1}^{1:A}, \phi))$$

$$\mathbf{S}_t^a = \mu_\theta^a(\mathbf{S}_{1:t-1}^{1:A}, \phi) + \sigma_\theta^a(\mathbf{S}_{1:t-1}^{1:A}, \phi) \cdot \mathbf{Z}_t^a \qquad \mathbf{Z}_t^a \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$$



Joint rollouts

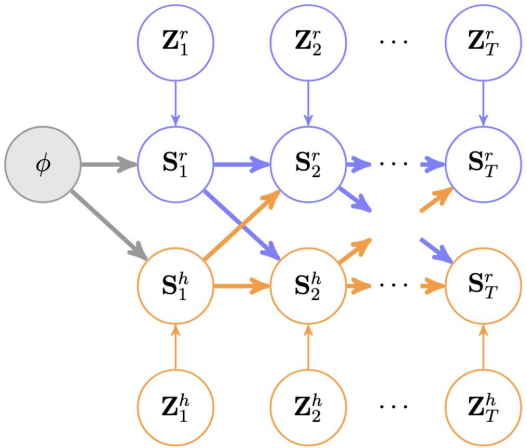
- State of agent *a* at time *t* depends on previous states of all agents
- Model learns: $p(\mathbf{S}_t^a \mid \mathbf{S}_{1:t-1}^{1:A}, \phi)$
- Full probability model: $p(\mathbf{S} \mid \phi) = \prod_{t=1}^{T} \prod_{a=1}^{A} p(\mathbf{S}_t^a \mid \mathbf{S}_{1:t-1}^{1:A}, \phi)$

ϕ Placeholder for conditional data

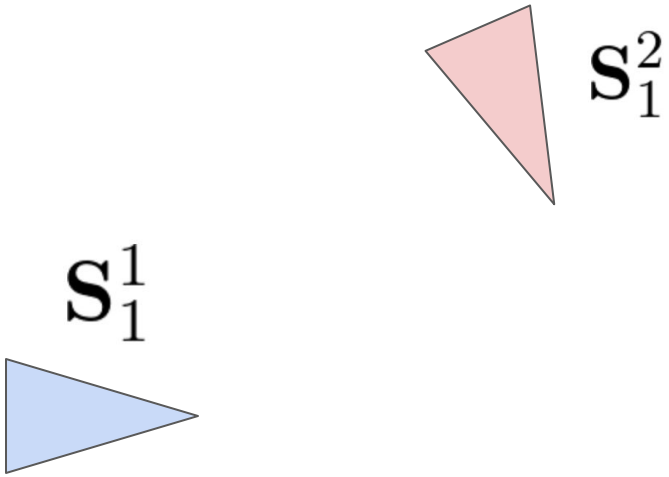
$$p(\mathbf{S}_t^a \mid \mathbf{S}_{1:t-1}^{1:A}, \phi) = \mathcal{N}(\mathbf{S}_t^a; \mu_\theta^a(\mathbf{S}_{1:t-1}^{1:A}, \phi), \Sigma_\theta^a(\mathbf{S}_{1:t-1}^{1:A}, \phi))$$

$$\mathbf{S}_t^a = \mu_\theta^a(\mathbf{S}_{1:t-1}^{1:A}, \phi) + \sigma_\theta^a(\mathbf{S}_{1:t-1}^{1:A}, \phi) \cdot \mathbf{Z}_t^a \qquad \mathbf{Z}_t^a \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$$

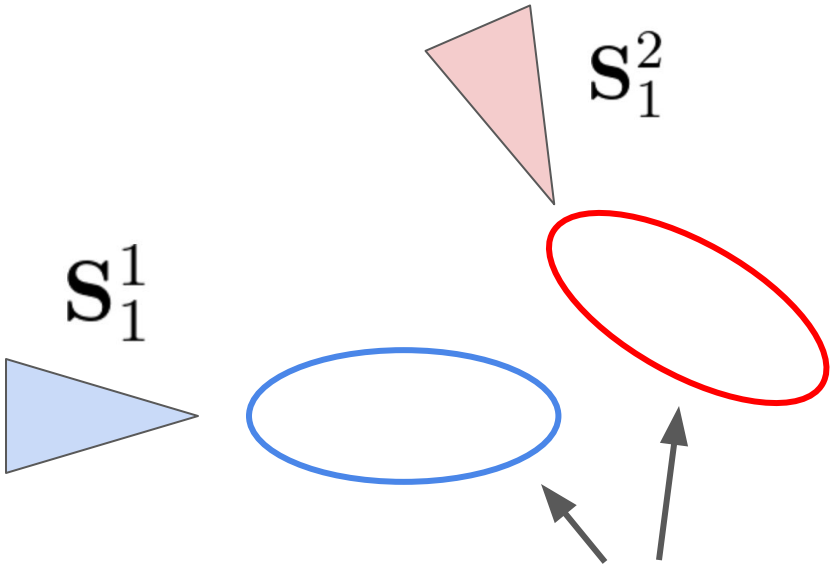
The learned part: predict mean and standard deviation for next state given all previous states



Joint rollouts

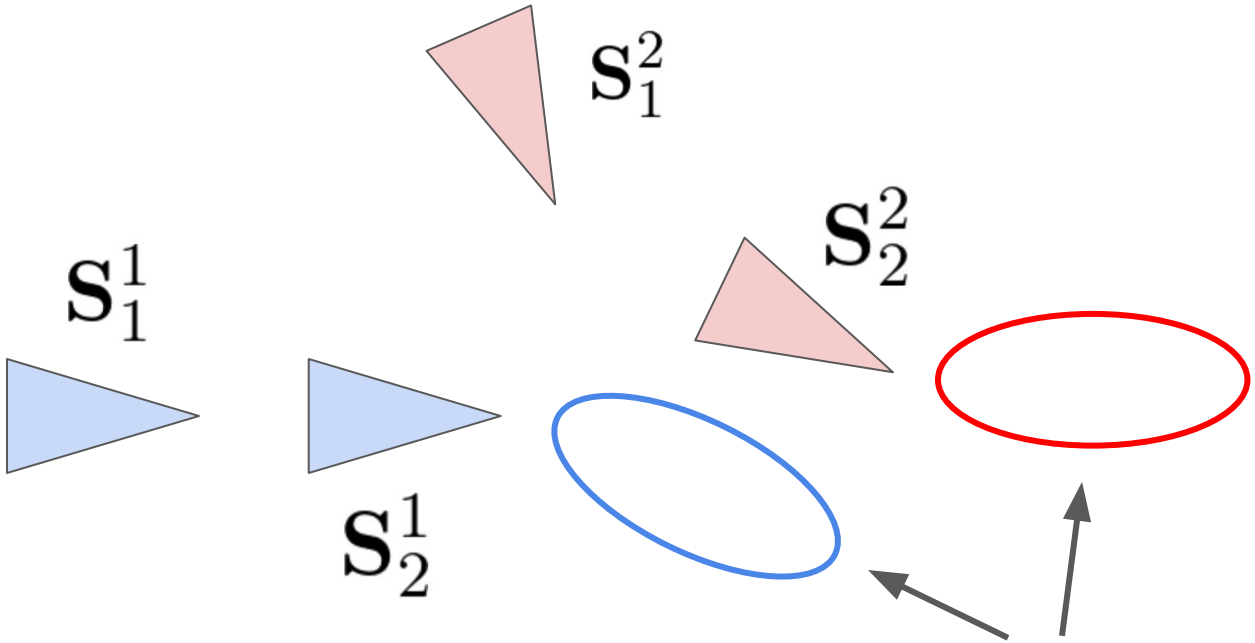


Joint rollouts



Model's predictions: gaussian for each agent at each timestep

Joint rollouts



Model's predictions: gaussian for each agent at each timestep

Training: loss function

- Maximum likelihood training
  - Feed in GT states for times 1…*t*-1
  - Predict mean and standard deviation for time *t*
  - Loss function is negative log likelihood of GT state at time *t* given this mean and stdev

Log likelihood of the data

Sum over single-timestep single-agent log likelihoods

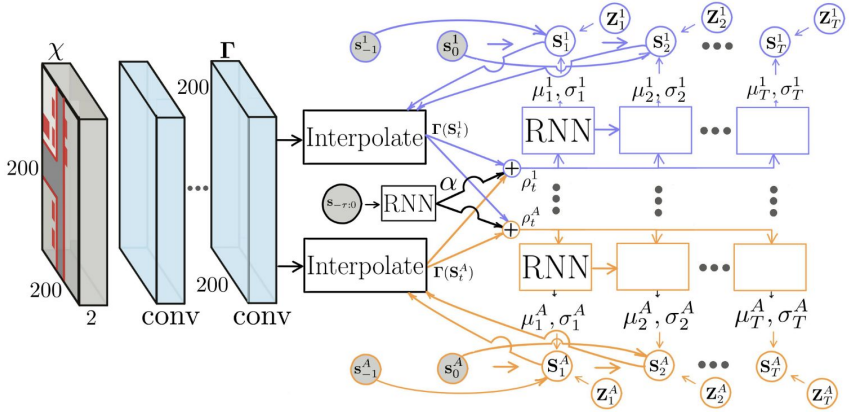
$$\log p(\mathbf{S}_{1:T}^{1:A} \mid \phi) = \sum_{t=1}^{T} \sum_{a=1}^{A} \log p(\mathbf{S}_t^a \mid \mathbf{S}_{1:t-1}^{1:A}, \phi)$$

$$= \sum_{t=1}^{T} \sum_{a=1}^{A} \log \mathcal{N}\left(\mathbf{S}_t^a; \mu_\theta^a(\mathbf{S}_{1:t-1}^{1:A}, \phi), \Sigma_\theta^a(\mathbf{S}_{1:t-1}^{1:A}, \phi)\right)$$

Gaussian log likelihood with predicted mean and variance

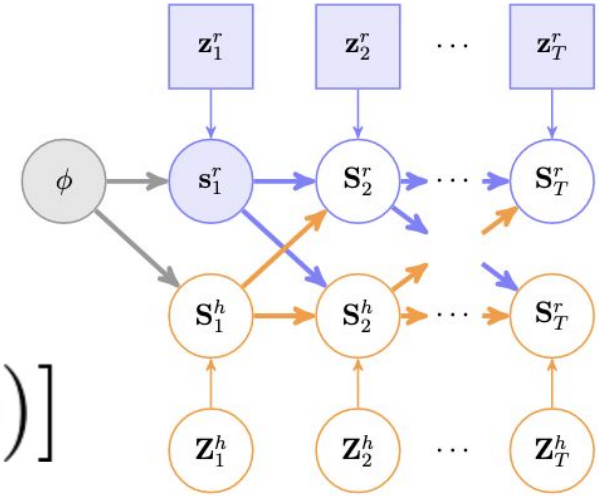
Training: architecture details (many no longer SOTA)

- Scene context: birds' eye view LIDAR data, processed by a CNN
- Interpolate at given position to get scene context feature vector
- GRU encoder for agent histories
- GRU maintains state for each agent during rollout
- Use verlet steps (a concept of a kinematics model)



PRECOG Planning

- Optimize over robot's Z variables



Robot's Z "actions"

Robot goal

$$\tilde{\mathcal{L}}(\mathbf{z}^r, \mathcal{G}) = \mathbb{E}_{\mathbf{Z}^h}\left[\log p(\mathcal{S} \mid \mathcal{G}, \phi)\right]$$

Expectation over human actions

Log probability of joint trajectory

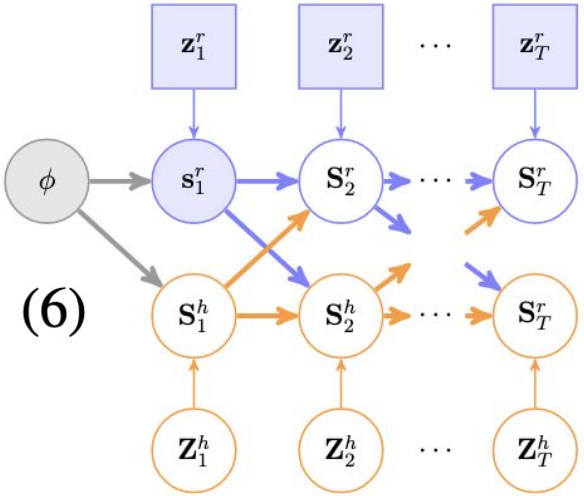
$$\mathbf{z}^{r*} = \text{argmax}_{\mathbf{z}^r}\, \mathcal{L}(\mathbf{z}^r, \mathcal{G}).$$

PRECOG Planning

- Evaluating log probability

$$\log \mathbb{E}_{\mathbf{Z}^h}\left[p(\mathbf{S}|\mathcal{G},\phi)\right] \;\geq\; \mathbb{E}_{\mathbf{Z}^h}\left[\log p(\mathbf{S}|\mathcal{G},\phi)\right] \tag{6}$$

Jensen's inequality



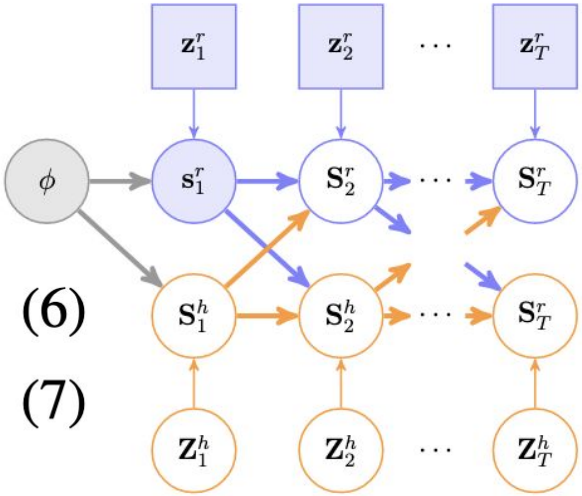
PRECOG Planning

- Evaluating log probability

$$\log \mathbb{E}_{\mathbf{Z}^h}[p(\mathbf{S}|\mathcal{G},\phi)] \;\geq\; \mathbb{E}_{\mathbf{Z}^h}[\log p(\mathbf{S}|\mathcal{G},\phi)] \qquad (6)$$

$$= \mathbb{E}_{\mathbf{Z}^h}[\log\left(q(\mathbf{S}|\phi)p(\mathcal{G}|\mathbf{S},\phi)\right)] - \log p(\mathcal{G}|\phi) \qquad (7)$$

Bayes' rule



$$p(\mathbf{S} \mid \mathcal{G}, \phi) = \frac{q(\mathbf{S} \mid \phi)p(\mathcal{G} \mid \mathbf{S}, \phi)}{p(\mathcal{G} \mid \phi)}$$

PRECOG Planning

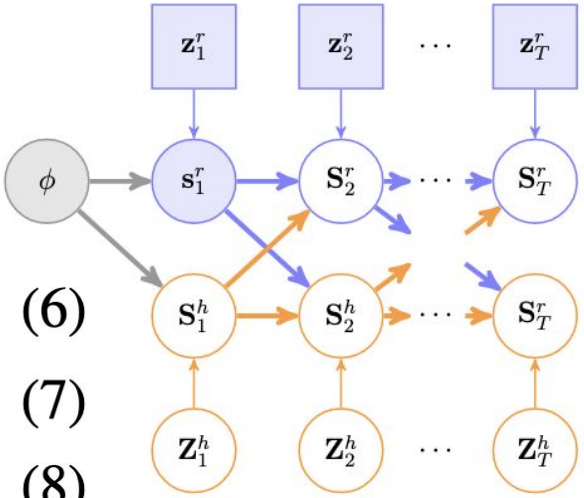
- Evaluating log probability

$$\log \mathbb{E}_{\mathbf{Z}^h}[p(\mathbf{S}|\mathcal{G},\phi)] \;\geq\; \mathbb{E}_{\mathbf{Z}^h}[\log p(\mathbf{S}|\mathcal{G},\phi)] \tag{6}$$

$$= \mathbb{E}_{\mathbf{Z}^h}[\log\left(q(\mathbf{S}|\phi)p(\mathcal{G}|\mathbf{S},\phi)\right)] - \log p(\mathcal{G}|\phi) \tag{7}$$

$$\mathcal{L}(\mathbf{z}^r,\mathcal{G}) \doteq \mathbb{E}_{\mathbf{Z}^h}[\log q(\mathbf{S}|\phi) + \log p(\mathcal{G}|\mathbf{S},\phi)] \tag{8}$$

Simplify



PRECOG Planning

- Evaluating log probability

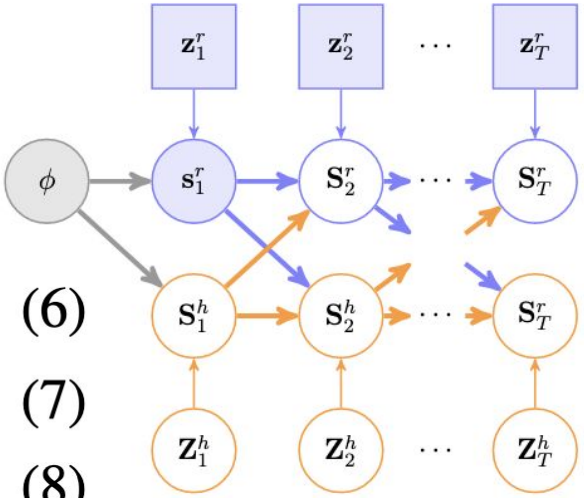
$$\log \mathbb{E}_{\mathbf{Z}^h}[p(\mathbf{S}|\mathcal{G},\phi)] \;\geq\; \mathbb{E}_{\mathbf{Z}^h}[\log p(\mathbf{S}|\mathcal{G},\phi)] \tag{6}$$

$$= \mathbb{E}_{\mathbf{Z}^h}[\log\left(q(\mathbf{S}|\phi)p(\mathcal{G}|\mathbf{S},\phi)\right)] - \log p(\mathcal{G}|\phi) \tag{7}$$

$$\mathcal{L}(\mathbf{z}^r,\mathcal{G}) \doteq \mathbb{E}_{\mathbf{Z}^h}[\log q(\mathbf{S}|\phi) + \log p(\mathcal{G}|\mathbf{S},\phi)] \tag{8}$$

$$= \mathbb{E}_{\mathbf{Z}^h}[\log \underbrace{q(f(\mathbf{Z})|\phi)}_{\text{multi-agent prior}} + \log \underbrace{p(\mathcal{G}|f(\mathbf{Z}),\phi)}_{\text{goal likelihood}}], \tag{9}$$

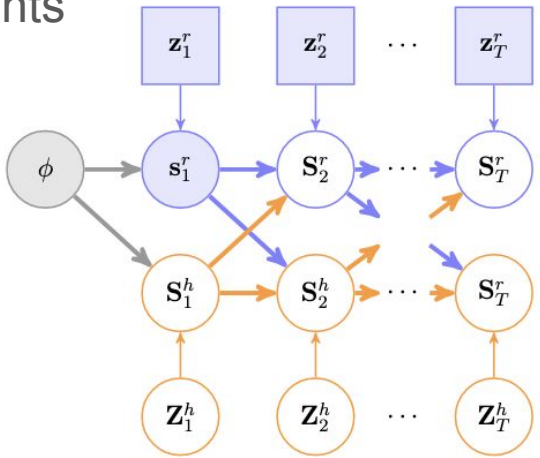
Write in terms of Z



$$\mathbf{S} = f_\theta(\mathbf{Z},\phi)$$

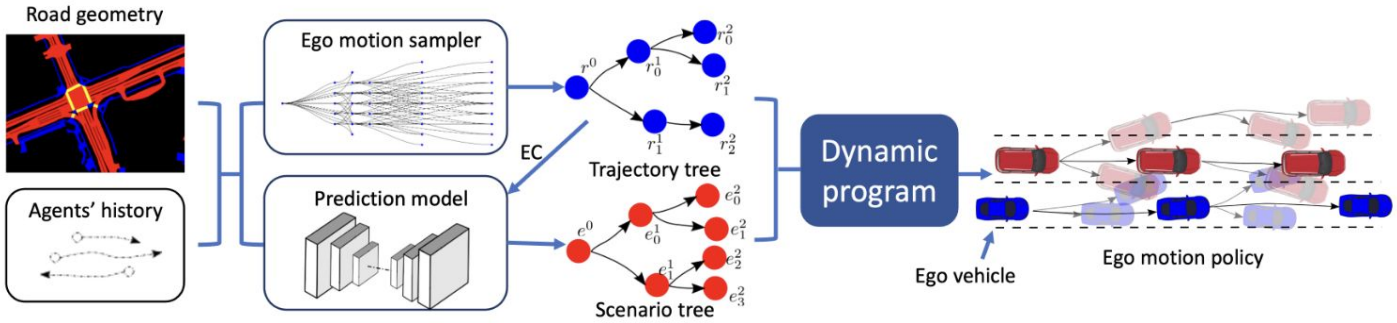
What is a PRECOG “plan”?

- Planner is selecting the Z’s for the robot
- Robot’s actual trajectory (i.e. the states) depend on:
  - Robot’s Z’s
  - The surrounding agents’ states
  - The learned parameters of the model
- Z’s represent a way of responding to surrounding agents
- They are not:
  - Actions for a kinematics model
  - An explicit sequence of states



Potential future work (a lot of it already done)

- Latent variables are not interpretable → discrete latent variables
- Probabilities are not enough → use cost terms for planning (e.g. collision)
- Modern architectures → transformers
- Richer per-step distributions → categorical over discrete grid of actions
- More complex planning algorithms → tree search



Archaeologist

Prior work: Motion forecasting for planning

[Multimodal Probabilistic Model-Based Planning for Human-Robot Interaction (2017)]

- Similarities
  - Conditions predictions on the robot's future trajectory
  - Rollout architecture with gaussian sampling at each timestep
- Differences
  - PRECOG makes joint predictions for multiple humans
  - PRECOG incorporates map context
  - PRECOG "robot plans" depend on other agents
  - PRECOG uses a causal factorization

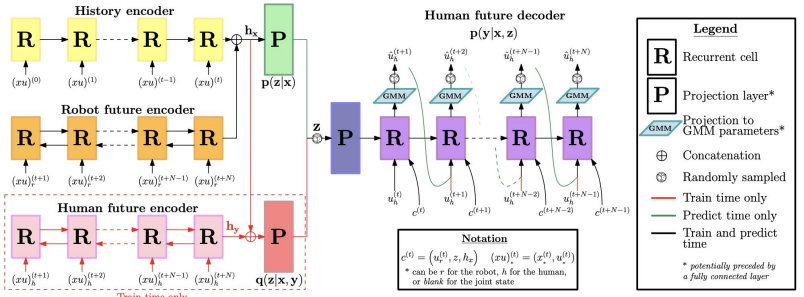
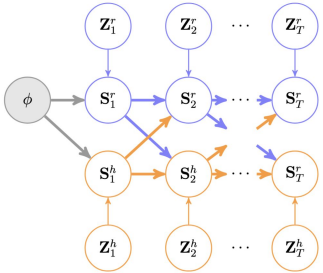


Fig. 2. CVAE architecture for sequence-to-sequence generative modeling of future human actions $\mathbf{y} = u_h^{(t+1:t+N)}$ conditioned on joint interaction history $(x^{(0:t)}, u^{(0:t)})$ and candidate robot future actions $u_r^{(t+1:t+N)}$ (together, $\mathbf{x}$). The random variable $\mathbf{z}$ is a latent mixture component index.

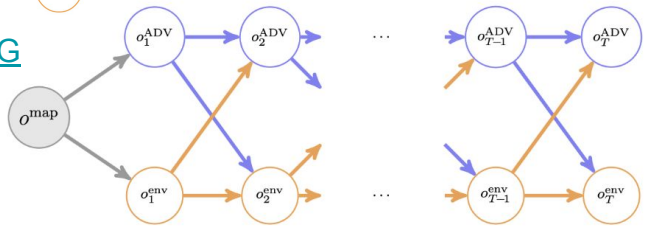
Subsequent work: Waymo Open Sim Agents Challenge

The Waymo Open Sim Agents Challenge (2023)

- Causal factorization: predictions for agent at time *t* should not depend on robot's plan for time *t+1* (conditioned on robot's plan at time *t*)

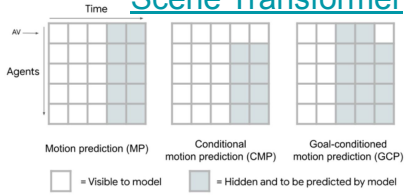


PRECOG

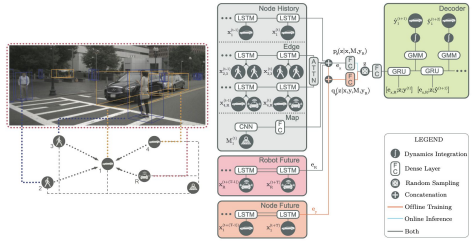


Waymo Sim Agents Challenge

Scene Transformer



Trajectron++



Multimodal Probabilistic Model-Based Planning for Human-Robot Interaction

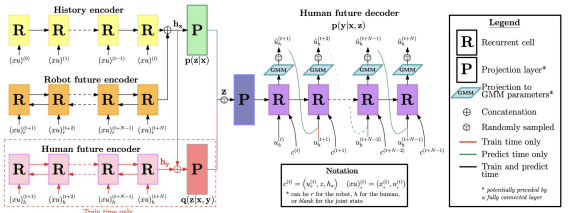


Fig. 2. CVAE architecture for sequence-to-sequence generative modeling of future human actions $\mathbf{y} = \mathbf{u}_h^{(t+1:t+N)}$ conditioned on joint interaction history $(\mathbf{x}^{(0:t)}, \mathbf{u}^{(0:t)})$ and candidate robot future actions $\mathbf{u}_r^{(t+1:t+N)}$ (together, $\mathbf{x}$). The random variable $\mathbf{z}$ is a latent mixture component index.

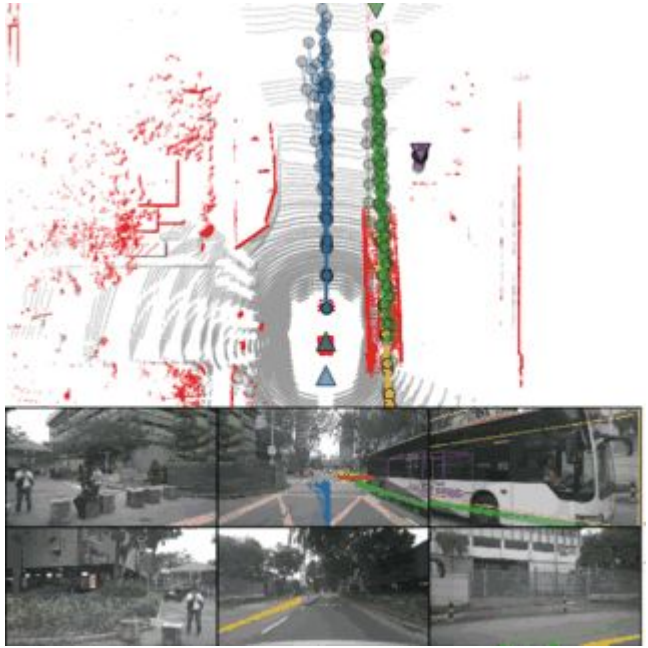
Subsequent work: Joint Rollout Architectures

- Multiple Futures Prediction (2019)
  - Discrete latent variables
  - Single latent variable for full trajectory
  - “Classmates forcing”
- LatentFormer: Multi-Agent Transformer-Based Interaction Modeling and Trajectory Prediction (2022)
  - Transformers
- MotionLM: Multi-Agent Motion Forecasting as Language Modeling (2023)
  - Discrete action space
- Tractable Joint Prediction and Planning over Discrete Behavior Modes for Urban Driving (2024)
  - Cost functions (e.g. collisions)

 Industry practitioner

Goal Conditioned Prediction

- PRECOG (PREdiction Conditioned On Goals).
- Objective: Use this probabilistic forecasting model to predict and influence agent interactions by conditioning forecasts on desired agent goals.
- Outcome: More reliable control in multi-agent settings by anticipating agent behaviors and ensuring safer interactions.
- Importance: Resolves difficulties we currently face in complex scenarios like intersections.



Issues with Prior approaches

- Agent interaction complexity: Real-world multi-agent interactions (e.g., vehicles, robots) are difficult to model as each agent influences others.
- Limited foresight: Previous methods could only predict future states but struggled to model co-influences between agents effectively.
- Handling uncertainty: Previous models had limited capabilities for addressing the stochastic nature of human or other agents' behavior, leading to suboptimal planning.

How Does PRECOG Address It?

- Goal-Conditioned Forecasting: The paper introduces PRECOG, which conditions predictions on the goals of one or more agents, enabling planning and interaction modeling.
- Factorized Latent Variables: Allows the system to model dependencies between agents by separating and updating each agent's predictions based on the actions of others.
- Multi-Agent Imitative Planning: Combines expert demonstrations and a probabilistic model to balance goal achievement with realistic agent behavior.
- Evidence: Outperforms state-of-the-art methods in multi-agent forecasting and planning on real-world datasets (CARLA, nuScenes) with substantial accuracy improvements.

Can we use it? Is it Scalable?

- Likelihood based multi-agent forecasting model as planner.
- Similar to our current architecture of the planning stack.

- Supports both fixed and dynamic agent counts:
  - Fixed count model
  - Partial Flexible count model
  - Fully Flexible count model
- The paper hints at extensions for multi-AV coordination, suggesting broader applications like fleet control.
- Ongoing improvements in conditional forecasting will improve real-time decision-making, hence clear potential.

Conclusion

Core Contributions: Test of Time

- Accept! (if we're back in 2019)
- Joint conditional motion forecasting
  - Still very popular
  - Modern approaches use more explicit dynamics / action spaces
- Goal conditioning
  - Very common, both for ego planning and other agent motion forecasting
  - More often done as a conditional input to the model, not Bayes' rule
- Probabilistic approach to planning
  - Argmax over robot actions, expectation over human actions is a common formulation
  - Modern approaches tend to have a separation between agent and ego policies
  - Modern approaches will have explicit cost terms

Discussion