WHAT STARTS HERE CHANGES THE WORLD

MAY 2021

LEARNING INVERSE KINODYNAMICS FOR ACCURATE HIGH-SPEED OFF-ROAD NAVIGATION ON UNSTRUCTURED TERRAIN

Xuesu Xiao[1], Joydeep Biswas[1], and Peter Stone[1,2]

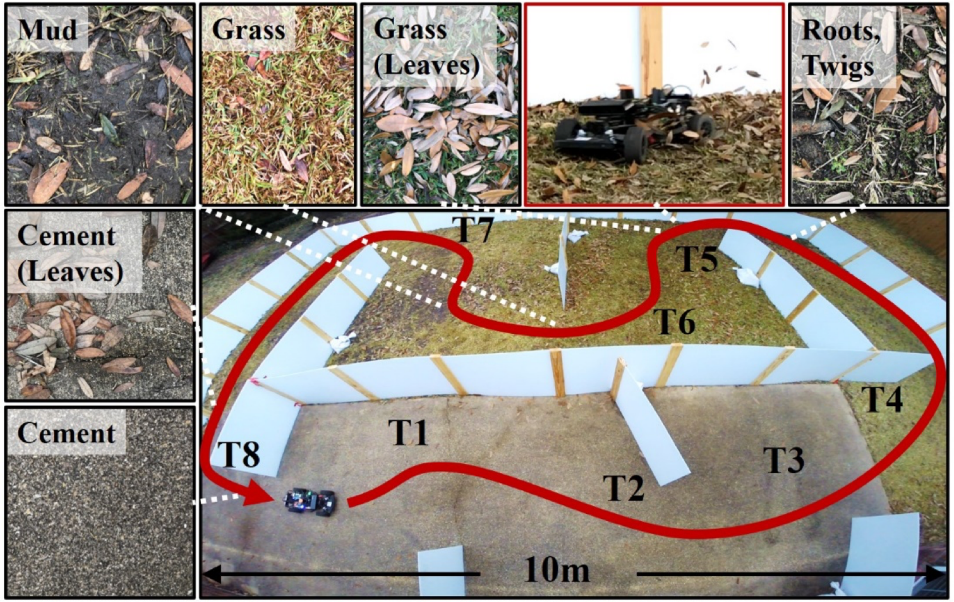
[1]The University of Texas at Austin, [2]Sony AI

JOYDEEP BISWAS

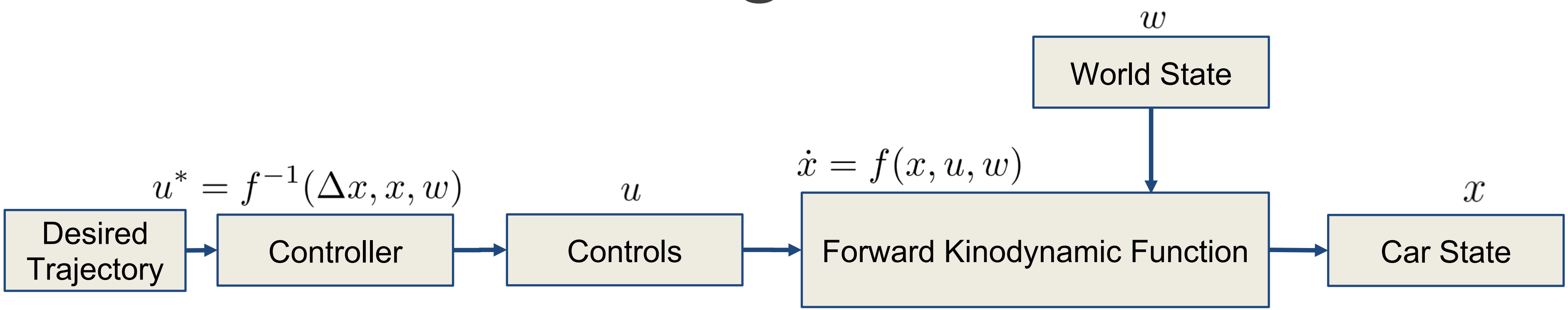
Assistant Professor, The University of Texas at Austin

Motivation

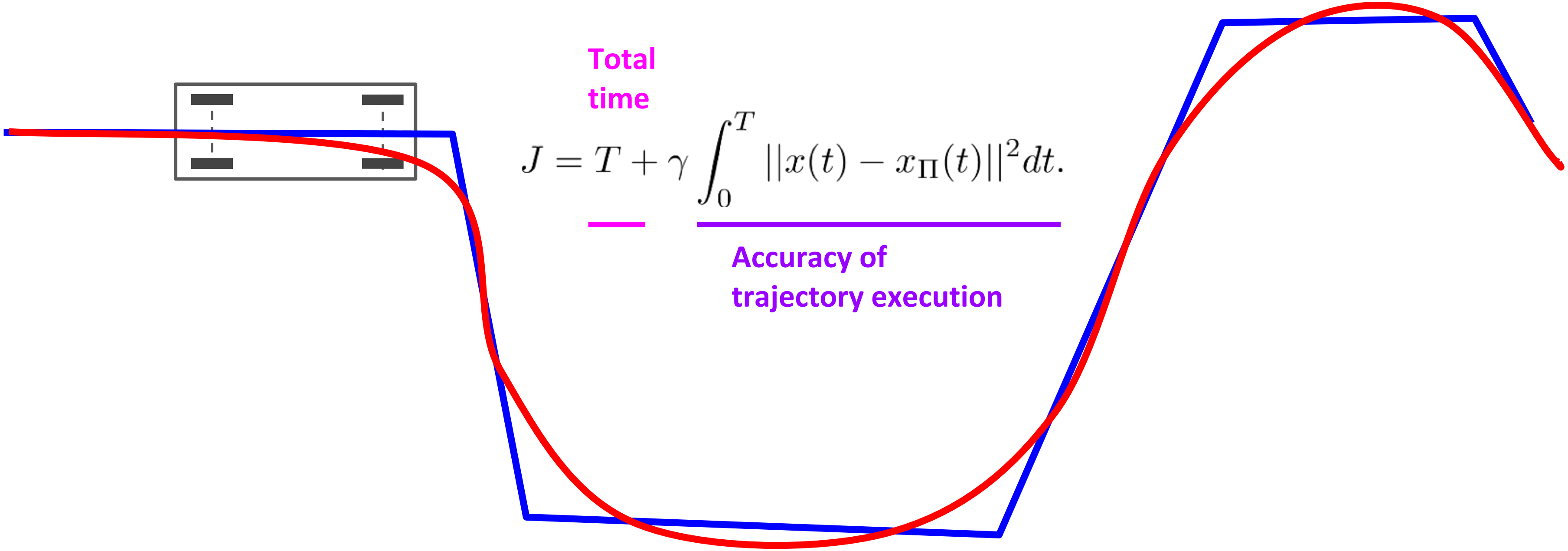
- Navigation becomes challenging under three combined conditions
  - Accurate
  - High-Speed
  - Off-Road (Unstructured Terrain)



Problem Setting



Controller Objective



Total time

$$J = \underbrace{T}_{\text{Total time}} + \gamma \underbrace{\int_0^T ||x(t) - x_{\Pi}(t)||^2 dt.}_{\text{Accuracy of trajectory execution}}$$

Accuracy of trajectory execution

Challenges With Off-Road Driving

Problem: Forward kino-dynamic function depends on unknown world states!

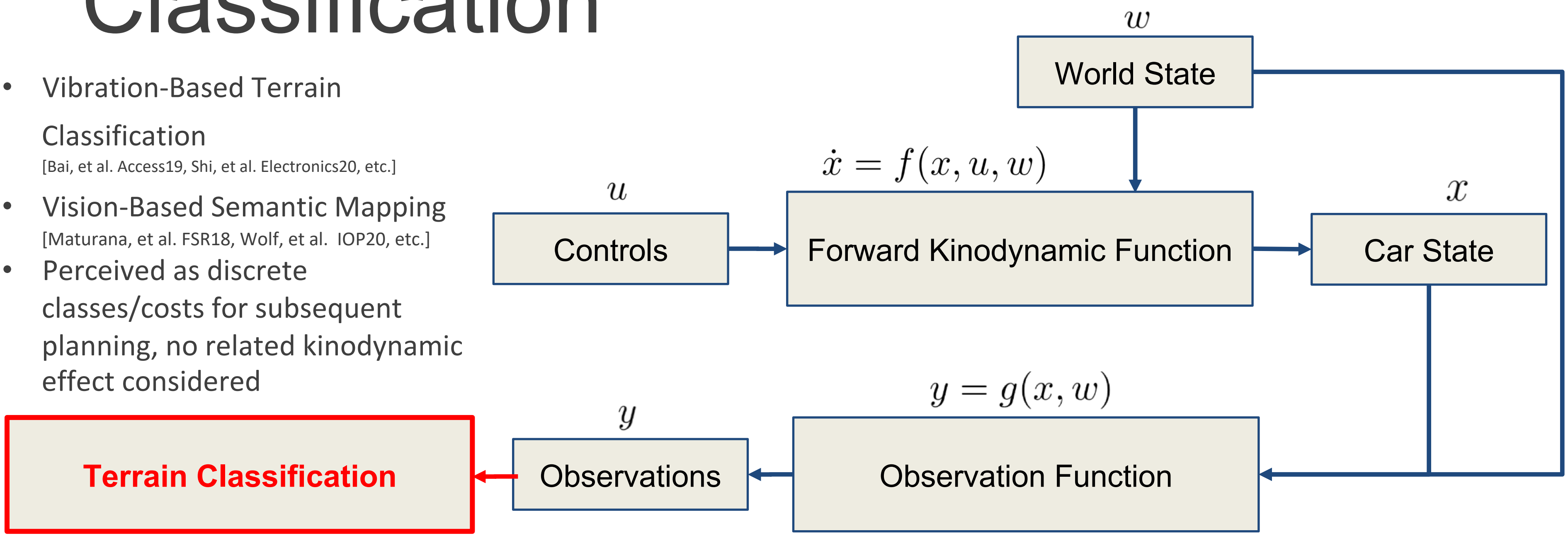
w
World State

$u^* = f^{-1}(\Delta x, x, w)$ $\qquad u \qquad$ $\dot{x} = f(x, u, w)$ $\qquad x$

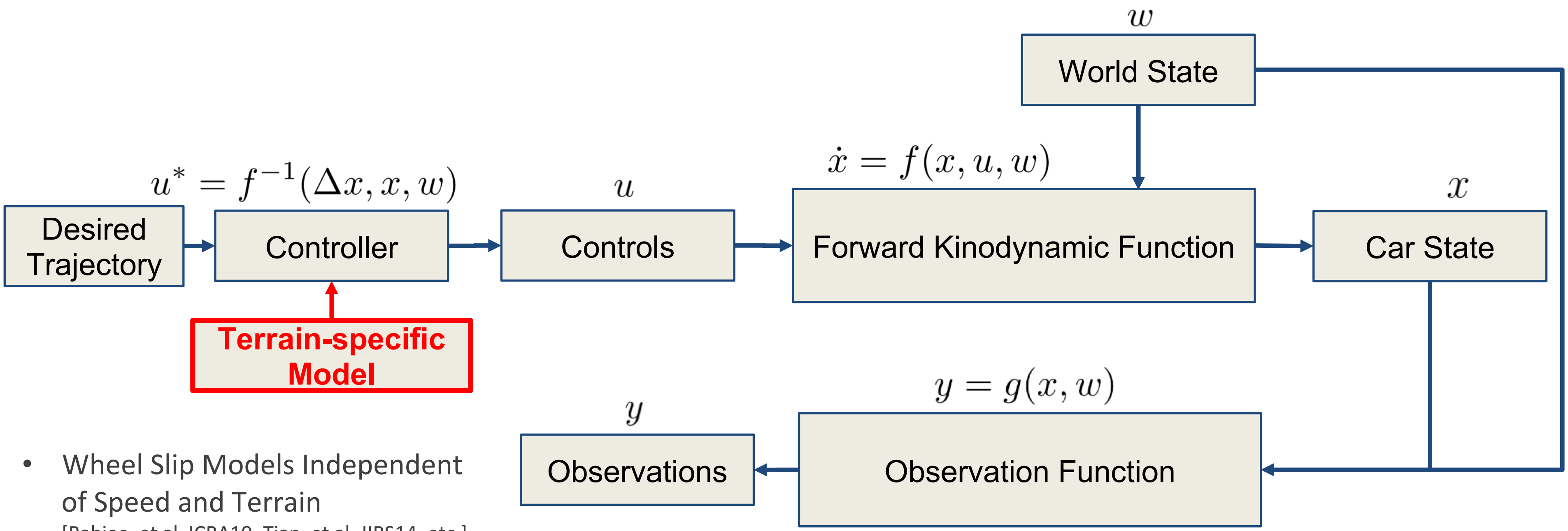
Desired Trajectory → Controller → Controls → Forward Kino-Dynamic Function → Car State

Related Work: Terrain Classification

- Vibration-Based Terrain Classification
  [Bai, et al. Access19, Shi, et al. Electronics20, etc.]
- Vision-Based Semantic Mapping
  [Maturana, et al. FSR18, Wolf, et al. IOP20, etc.]
- Perceived as discrete classes/costs for subsequent planning, no related kinodynamic effect considered



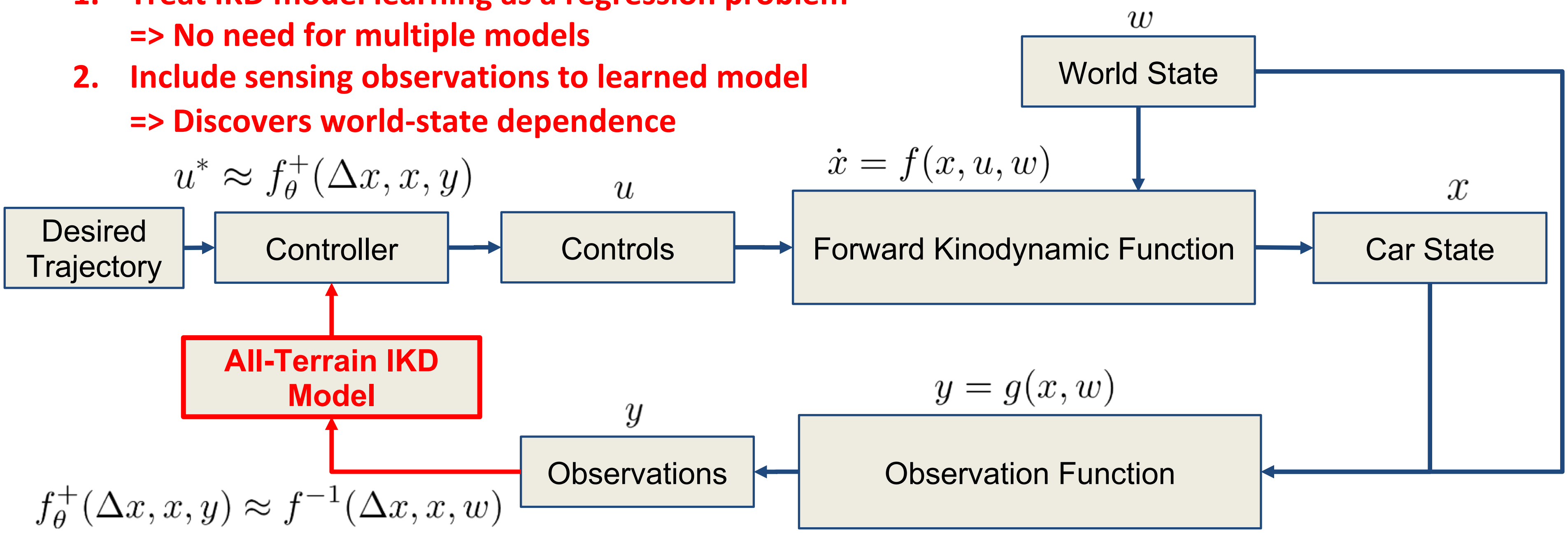
Related Work: Terrain-Specific Models



- Wheel Slip Models Independent of Speed and Terrain
  [Rabiee, et al. ICRA19, Tian, et al. JIRS14, etc.]
- End-to-End Machine Learning
  [Pan, et al. IJRR20, Siva, et al. RSS19]

Our Approach: Learning All-Terrain Inverse Kinodynamics

1. **Treat IKD model learning as a regression problem**
   => No need for multiple models
2. **Include sensing observations to learned model**
   => Discovers world-state dependence



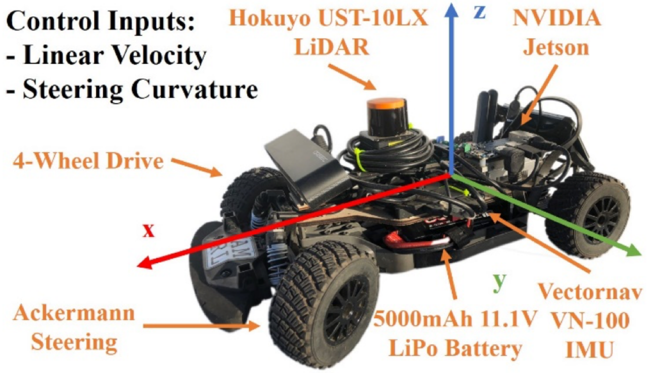
Training From “Off-Track Time”

1. Manually drive around the car on a variety of terrain types
   Collect:
   a. Joystick controls u^i
   b. State of the car x^i
   c. Observations y^i
   d. Actual outcomes from real-world forward kinodynamics Δx^i

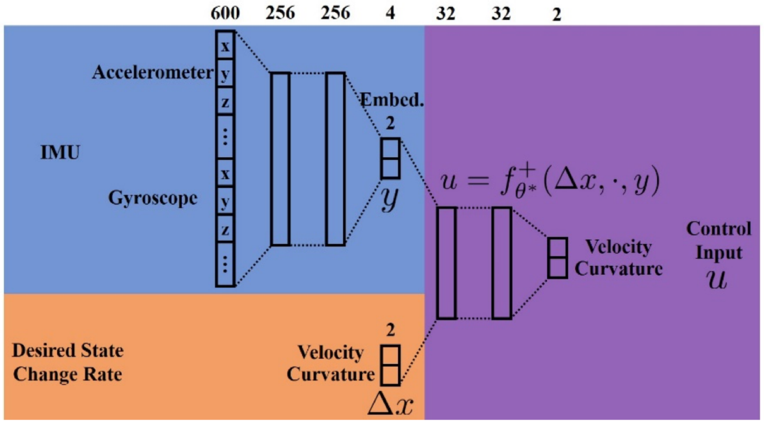
2. Train regression model with this as supervised loss:
   Pretend actual outcomes were desired, regression model should output the joystick controls

$$
\begin{aligned}
\theta^* &= \arg\min_{\theta} \sum_{(\Delta x^i, x^i, y^i) \in \mathcal{T}} \| f^{-1}(\cdot, \cdot, \cdot) - f_\theta^+(\Delta x^i, x^i, y^i) \|_H \\
&= \arg\min_{\theta} \sum_{(u^i, \Delta x^i, x^i, y^i) \in \mathcal{T}} \| u^i - f_\theta^+(\Delta x^i, x^i, y^i) \|_H ,
\end{aligned}
$$

Implementation



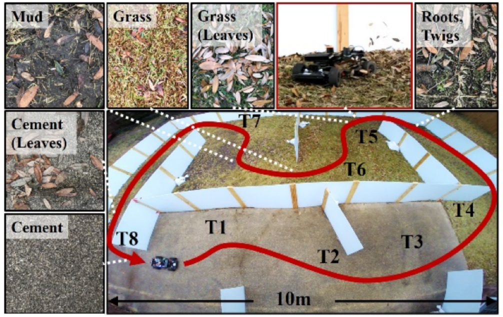
UT Automata

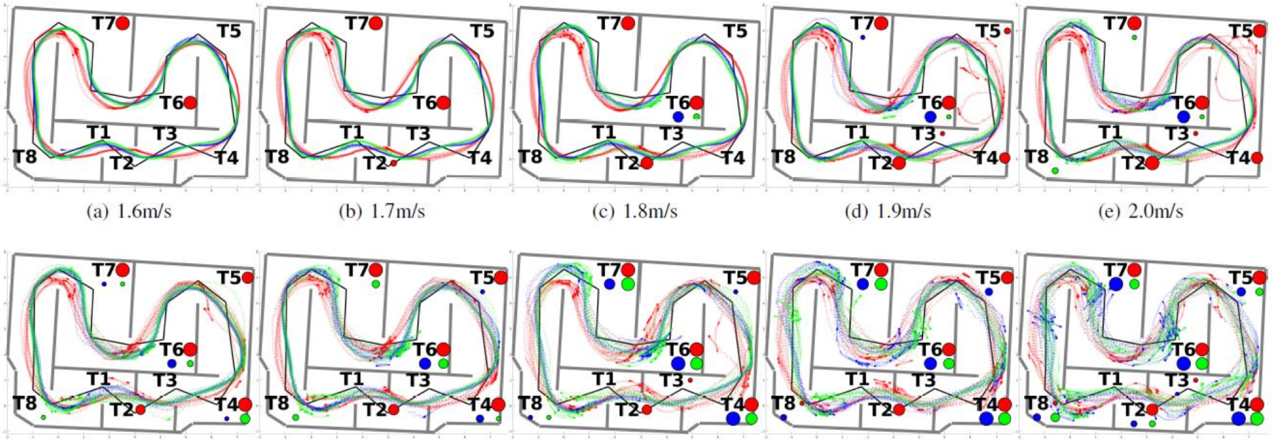


Neural Network Architecture

Experiment Results

Seen Terrain, Unseen Track





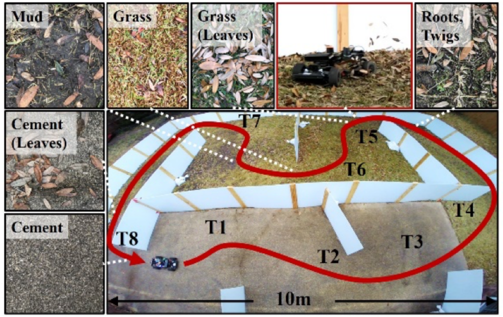
(a) 1.6m/s (b) 1.7m/s (c) 1.8m/s (d) 1.9m/s (e) 2.0m/s

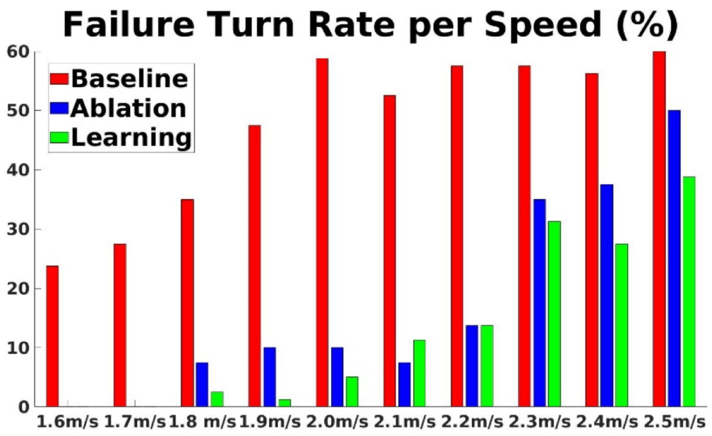
(f) 2.1m/s (g) 2.2m/s (h) 2.3m/s (i) 2.4m/s (j) 2.5m/s

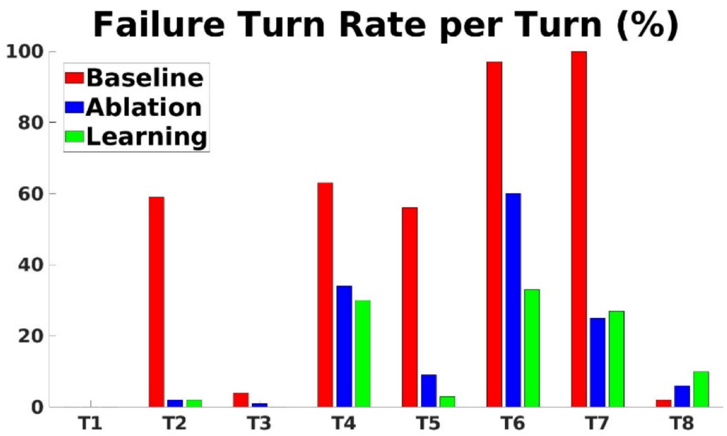
Red (Baseline) : No learned model
Blue (Ablation) : Learned model, no sensing inputs
Green (Ours) : Learned model with sensing inputs

Experiment Results

- Seen Terrain, Unseen Track

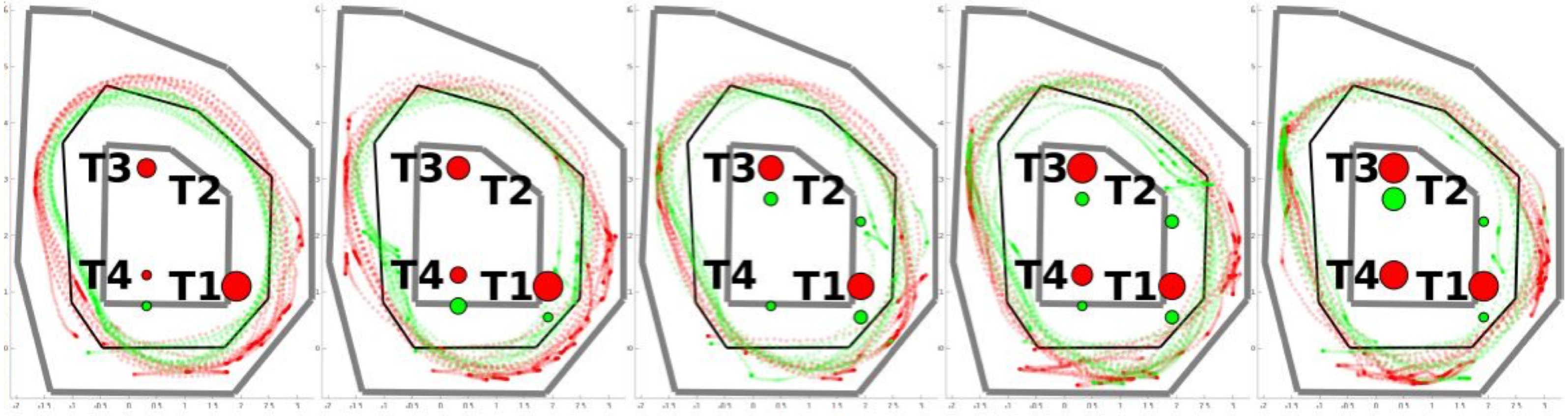






Experiment Results

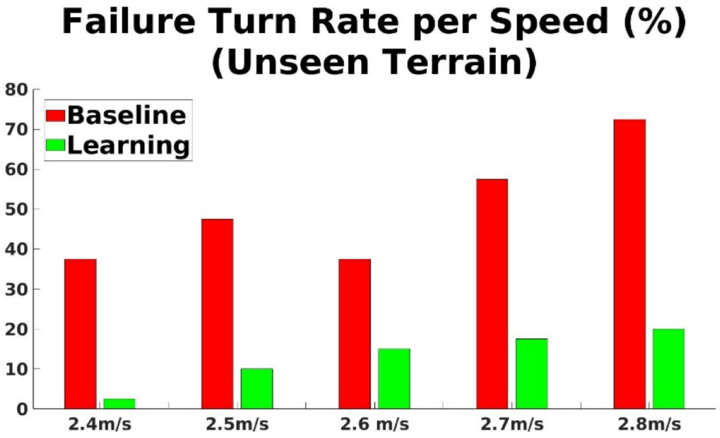
- Unseen (Easier) Terrain, Unseen Track

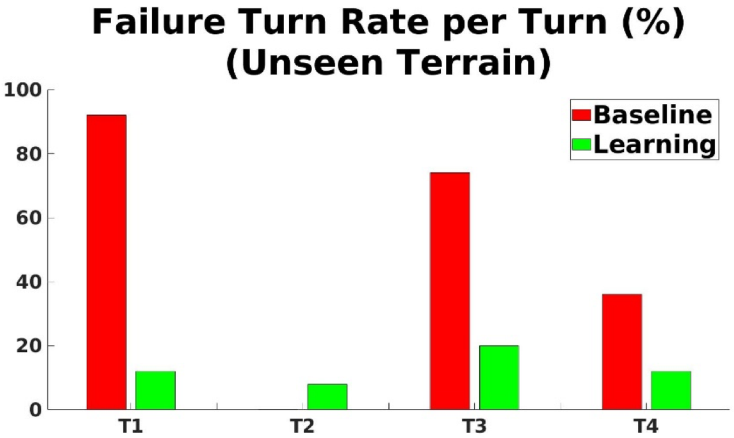


(b) 2.4m/s (c) 2.5m/s (d) 2.6m/s (e) 2.7m/s (f) 2.8m/s

Experiment Results

- Unseen (Easier) Terrain, Unseen Track





Conclusions

- Using inertia-based observation embeddings to capture elusive and stochastic world state during off-road navigation on unstructured terrain
- Learning inverse kinodynamic model for accurate and high-speed navigation in a data-driven manner
- Improving navigation performance in seen/unseen terrain and track layout
- Future Work
  - Adding vision-based observation to prepare for future wheel-terrain interactions
  - Generalization from easier to harder environments