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

Xuesu Xiao¹, Joydeep Biswas¹, and Peter Stone¹,²
¹The University of Texas at Austin, ²Sony 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

\[ u^* = f^{-1}(\Delta x, x, w) \]

\[ \dot{x} = f(x, u, w) \]

- Desired Trajectory
- Controller
- Controls
- Forward Kinodynamic Function
- World State
- Car State
Controller Objective

\[ J = T + \gamma \int_0^T \| x(t) - x_{\Pi}(t) \|^2 dt. \]
Challenges With Off-Road Driving

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

\[ u^* = f^{-1}(\Delta x, x, w) \]

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

\[ \dot{x} = f(x, u, w) \]
Related Work: Terrain Classification

- Vibration-Based Terrain Classification

- 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 Classification

\[ \dot{x} = f(x, u, w) \]

\[ y = g(x, w) \]
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

\[ u^* \approx f^+_\theta(\Delta x, x, y) \]

\[ \dot{x} = f(x, u, w) \]

\[ y = g(x, w) \]
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 $\Delta x^i$

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

\[
\theta^* = \arg\min_{\theta} \sum_{(\Delta x^i, x^i, y^i) \in T} \| f^{-1}(\cdot, \cdot, \cdot) - f_{\theta}^+(\Delta x^i, x^i, y^i) \|_H
\]

\[
= \arg\min_{\theta} \sum_{(u^i, \Delta x^i, x^i, y^i) \in T} \| u^i - f_{\theta}^+(\Delta x^i, x^i, y^i) \|_H,
\]
Implementation

Control Inputs:
- Linear Velocity
- Steering Curvature

4-Wheel Drive

Ackermann Steering

Hokuyo UST-10LX LiDAR

NVIDIA Jetson

5000mAh 11.1V LiPo Battery

Vectornav VN-100 IMU

Embedded

UT Automata

Neural Network Architecture
Experiment Results

Seen Terrain, Unseen Track

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
Experiment Results

- Unseen (Easier) Terrain, Unseen Track

**Failure Turn Rate per Speed (%)**
(Unseen Terrain)

**Failure Turn Rate per Turn (%)**
(Unseen Terrain)
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