Literature Review: Physics-based AI

How have PINNs impacted the physics-based AI field?

Resource: Physics-Based Deep Learning Book

- Maintained by Thuerey Group (Physics-based Simulation Group) at the Technical University of Munich
- 200-pg document summarizing several techniques for deep learning for physics
- Extensive coverage of using equation residuals for training NNs (PINN approach)
- https://arxiv.org/pdf/2109.05237.pdf
- https://physicsbaseddeeplearning.org/intro.html

Review: “Learning Motion Primitives Automata for Autonomous Driving Applications”

Review: “Learning Motion Primitives Automata for Autonomous Driving Applications”

- Motion planning for autonomous machines
  - selecting “primitives” to inform future actions taken by machine
- Traditional approaches for selecting primitives
  - learning from only data, learning via demonstration, numerical approximation etc.
  - Authors use PINNs to propose greater efficiency

PINN Approach

- Learning primitives from dataset and dynamical models for maneuver automation (MA)
  - Develop a PINN model
- This has applications in autonomous robot systems and self-driving cars
- Learning 2 types of primitives
  - Trim primitives - controlled relative equilibria
  - Maneuver primitives - controlled trajectories starting and ending on trims (connects two trim prims)

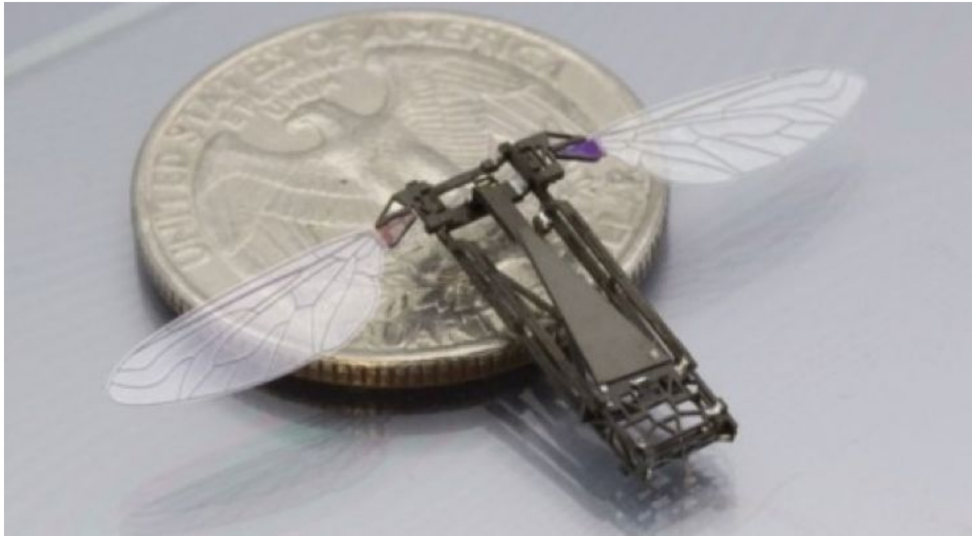
Simulation

- CommonRoad Benchmark
- Generate a MA as a directed graph
- Experiment with several “robots” (models) with varying properties
  - PINN architecture combines training data and dynamical laws for learning primitives
- This approach outperforms comparable “handcrafted” agents, demonstrating plausibility for improved autonomous driving performance

Review: “Efficient Modeling of Morphing Wing Flight Using Neural Networks and Cubature Rules”

Review: “Efficient Modeling of Morphing Wing Flight Using Neural Networks and Cubature Rules”

- Micro aerial vehicles (MAVs) - small unmanned aerial vehicles
    - May be inspired by insects

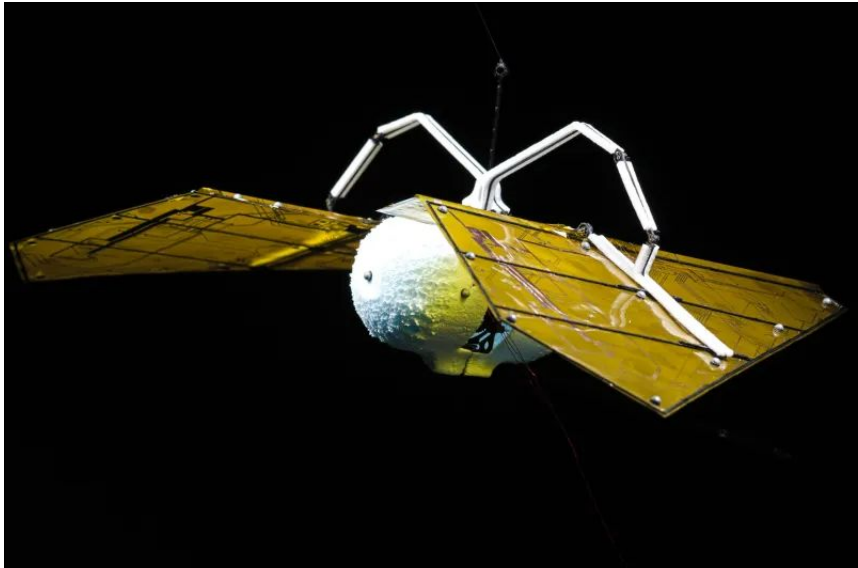


Motion of MAVs

- Case: “Morphing” MAVs - complex dynamics, wings constituting large part of mass
    - Dependence on Coriolis force, gravity, inertia, etc.
- Goal - estimate complex robotic models through machine learning techniques

Aerobat

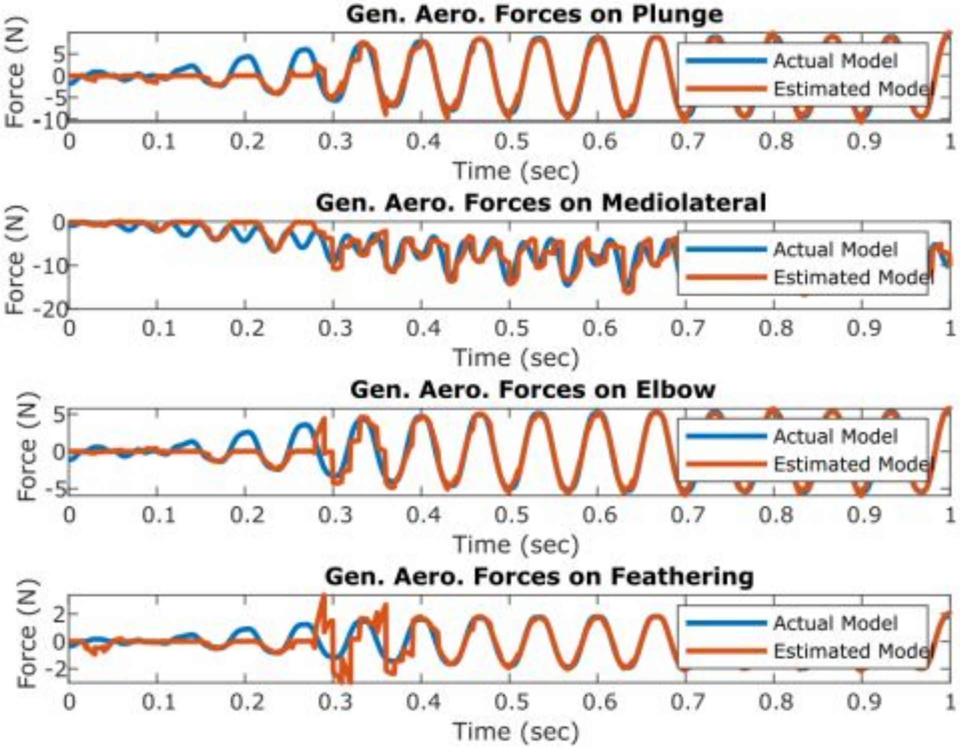
- A morphing MAV inspired by bats
- Developed in a Northeastern project
- Some components are difficult to model
  - We can’t fit traditional dynamics



PINNs

- Use PINNs to capture dynamics of MAVs
    - Increased accuracy and efficiency in modeling of robot-environment interactions

Simulation Results



Review: "On Learning the Dynamical Response of Nonlinear Control Systems with Deep Operator Networks"

Review: “On Learning the Dynamical Response of Nonlinear Control Systems with Deep Operator Networks”

- DeepONet Usage
- Tasks such as control, optimization, etc. are expensive for complex nonlinear systems
- Use DeepONet to learn operator mappings
    - Favorable given DeepONet’s promising performance on such complex equations
- Paper Objectives
    - A deep learning framework for mapping current state of control system and control input to the next state
    - Simulate a system’s response to control input

Method

- Use DeepONet to learn operators from data, then make predictions for future states of control systems
- Benchmark - explicit RK scheme

Experiments

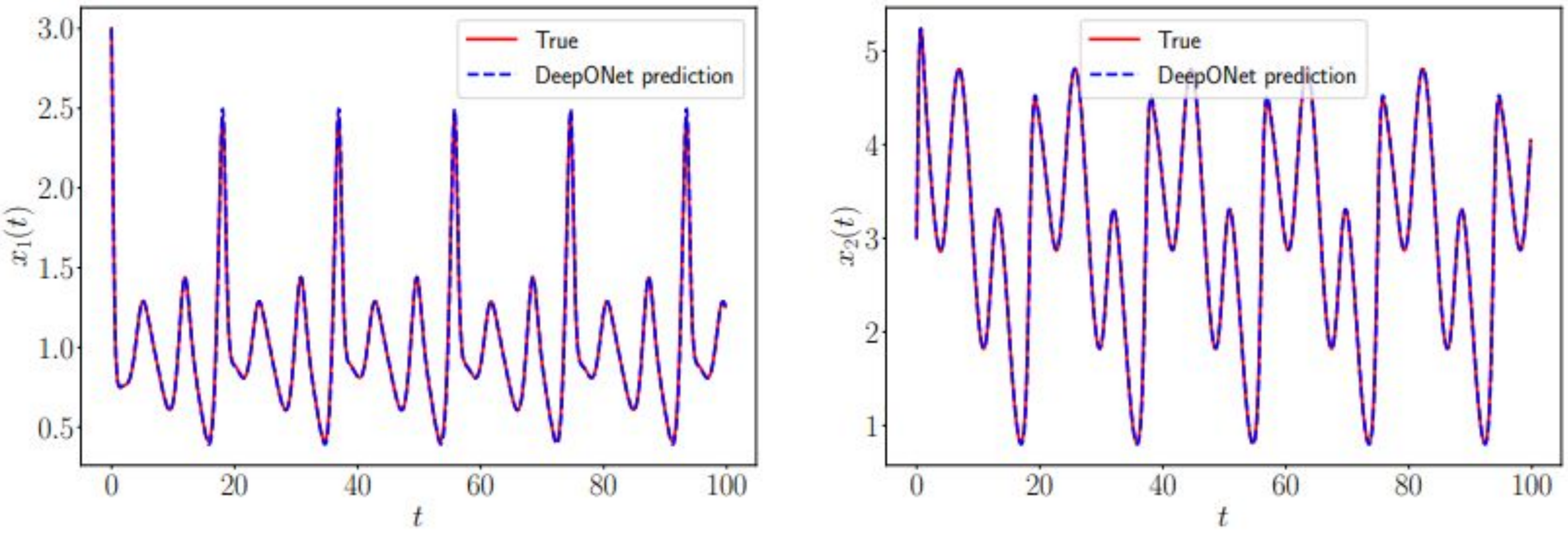
- Common control tasks, frequently used in RL control
- Predator-prey with control
- Pendulum swing-up
- Cart-pole

$$\dot{x}_1 = x_1 - x_1 x_2 + u(t)$$
$$\dot{x}_2 = -x_2 + x_1 x_2.$$

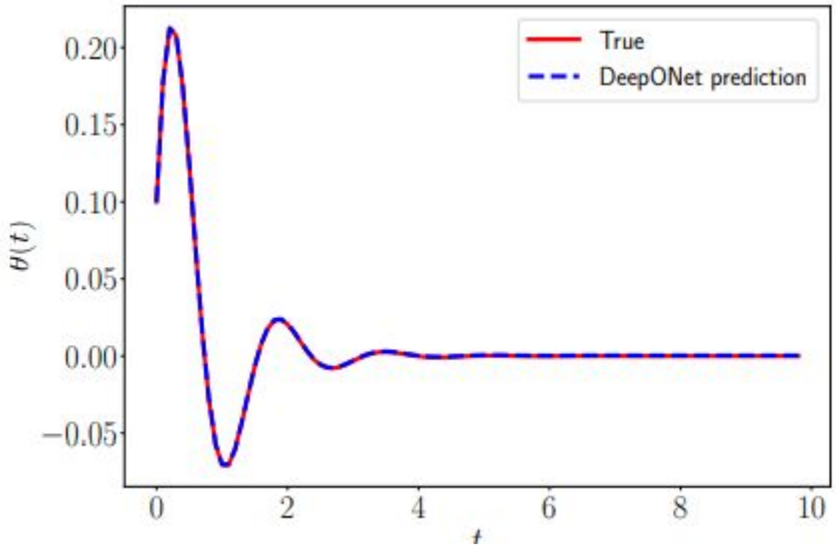
$$\ddot{\theta}\left(\frac{1}{4}ml^2 + I\right) + \frac{1}{2}mlg\sin\theta = u(t) - b\dot{\theta},$$

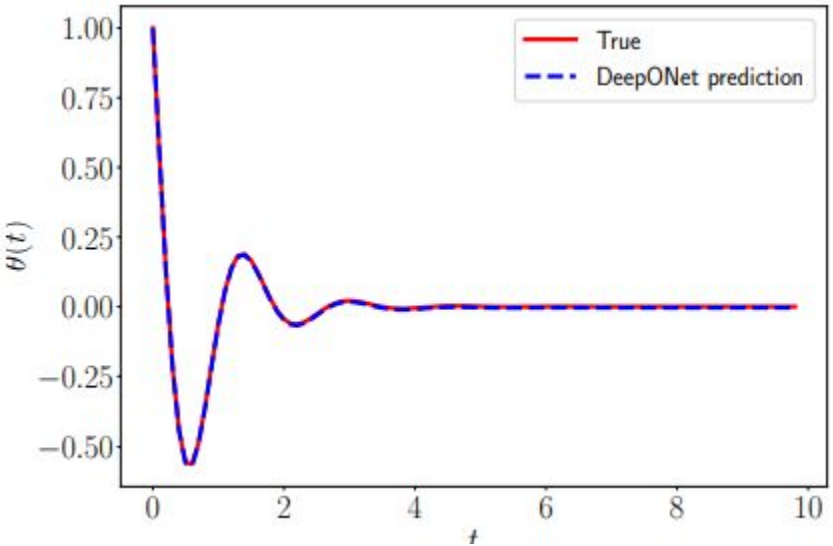
$$\ddot{\theta} = \frac{g\sin\theta + \cos\theta\left(\frac{-u(t) - m_p l\dot{\theta}^2\sin\theta}{m_c + m_p}\right)}{l\left(\frac{4}{3} - \frac{m_p\cos^2\theta}{m_c + m_p}\right)}$$
$$\ddot{p} = \frac{u(t) - b\dot{p} + m_p l(\dot{\theta}^2\sin\theta - \ddot{\theta}\cos\theta)}{m_c + m_p}.$$

Lotka-Volterra

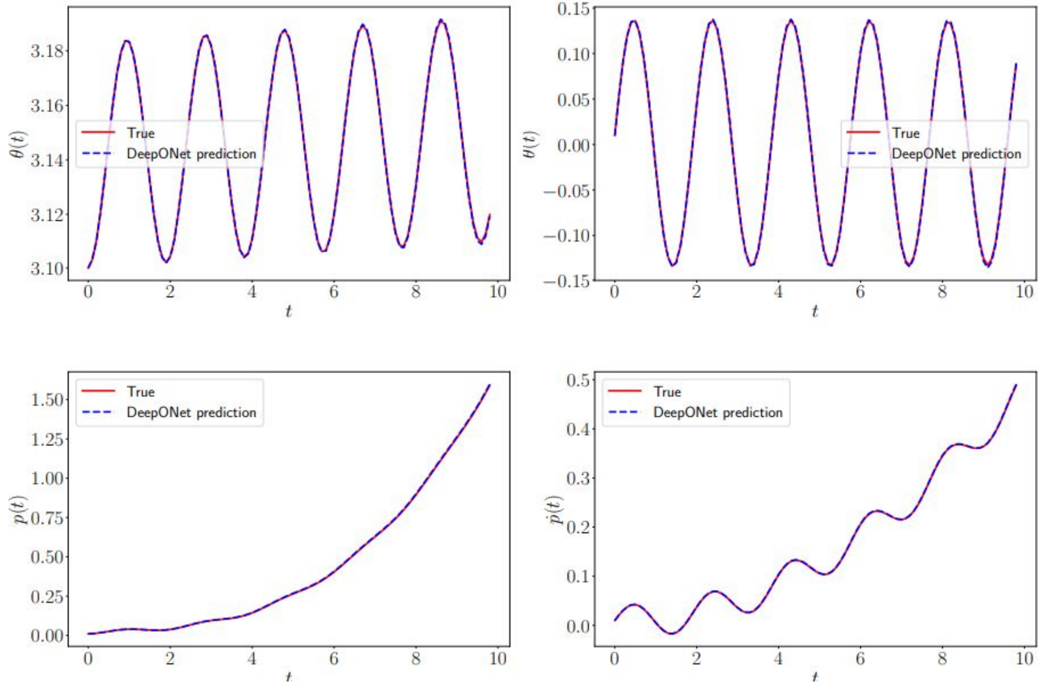


Pendulum Swing-Up





Cart-Pole



Areas to look

- DeepONets/PINNs for particular problems in mathematics/physics
  - Fluid dynamics, chaotic differential equations
- DeepONets/PINNs for robotics and control theory
  - PINNs have allowed for better performance in physics-informed learning tasks since their release