An Empirical Analysis of Methods for Learning Robot Kinematics from Demonstration

Northwestern University

Alexander Broad[1,2], Deepak Gopinath[1,2], Todd Murphey[1] and Brenna Argall[1,2]

[1]Northwestern University, Evanston, IL 60208

[2]Shirley Ryan AbilityLab, Chicago, IL, 60611

Motivation

- High-fidelity kinematic models can be used in the control of complex, high-dimensional robotic systems
- Errors in the kinematic model can result in many negative practical implications including instability of the system and unsafe control
- One way to improve upon pre-defined kinematic models is to learn the model directly from data
- In particular, learning from demonstration allows us to compute kinematic models based solely on observations of the robotic system interacting with the environment

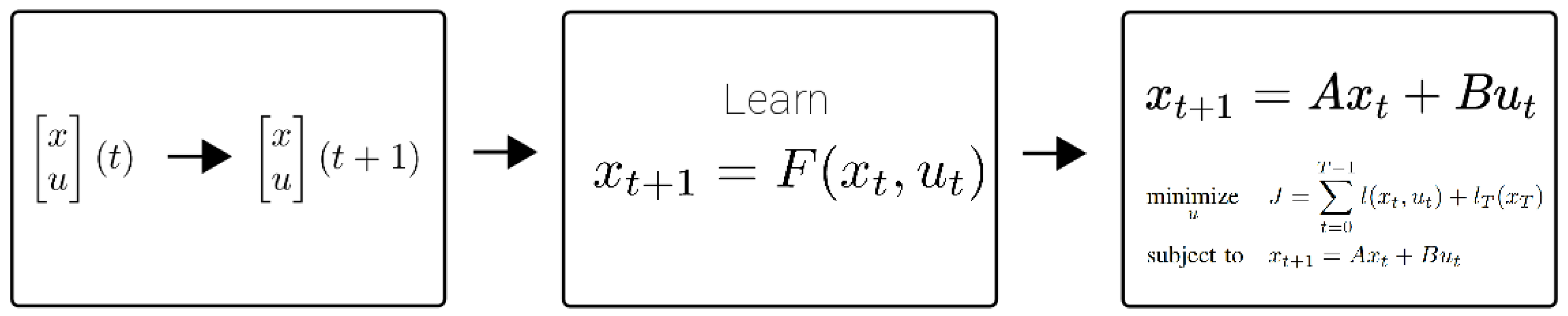


Fig. 1. Pictorial description of the proposed workflow

Model Learning Approaches

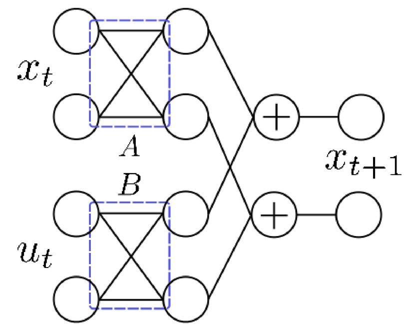
- In this work, we explore a number of modern approaches that can be used to learn models for robot kinematics directly from demonstration data
- We are specifically interested in learning models which are *actionable* in a Model Predictive Control framework (Fig. 1)
- Therefore, the model should be differentiable and interpretable by standard MPC frameworks, i.e.

$$\frac{\partial F}{\partial x} = A, \frac{\partial F}{\partial u} = B$$

- We are particularly interested in how these approaches
  1. handle low-data scenarios, and
  2. scale to high-dimensional data sources

Neural Networks

- **Neural Networks** (NN) [1] are a general learning approach for modeling linear/nonlinear functions
- To model linear systems, we can define the neural network structure based on the state space of the system, e.g.



- However, this approach will not work for nonlinear sys.
- Instead, we must use more complex network architectures and nonlinear activation functions
- In this case, we can compute the desired A & B matrices by taking numerical derivatives of the learned model

Koopman Operator

- The **Koopman operator** [2] is an infinite-dimensional linear operator that can capture all relevant information about any dynamical system

Standard Evolution Operator	Koopman Operator
$x_{t+1} = F(x_t)$	$\phi(x_{t+1}) = K\phi(x_t)$
Possibly Nonlinear?	Linear! / Hilbert Space Basis Function

Basis Function

$$\phi : \mathcal{M} \rightarrow \mathbb{C}$$

Original State Space — Hilbert Space

- Therefore, our learning method need not change depending on the linearity of the system kinematics
- Similar to the NN approach, we can then pair the linear models with linear MPC, and the nonlinear models with nonlinear MPC (e.g. iLQR [3], Sequential Action Control [4]), to generate policies online

Hardware Experiments

- Experimental Platform: **Kinova MICO** (Fig. 2)
- Linear representation: **End-effector** (EEF)
- Nonlinear representation: **Joint space**

Fig. 2. Kinova MICO

http://www.kinovarobotics.com/wp-content/uploads/2015/04/jaco-6D-2F-1.jpg

Preliminary Results

- In EEF space both, the Koopman operator and NNs learn effective models for use in an MPC framework
- Evaluation: **(1)** Ability of learned model + MPC to generate policies that successfully achieve the desired goal state **(2)** Robust to limited training data (Table 1)

Table 1.

	Goal Position 1		Goal Position 2		Goal Position 3	
# data pts.	Koopman	Neural Net	Koopman	Neural Net	Koopman	Neural Net
10						
100						
1000						
~12,000						

- Model + MPC succeeds   - Model + MPC fails

- When a model learns the system dynamics, results along the performance metrics are comparable
- Metrics: Distance to goal, Control effort, Path Length

Discussion and Future Work

- We observe a trend that suggests the Koopman operator is able to learn useful kinematic models from *less data* compared to Neural Networks
- We expect to find larger differences as we further explore the higher-dimensional representation of the robot system

References

1. H. Dinh, S. Bhasin, and W. E. Dixon. Dynamic Neural Network-Based Robust Identification and Control of a Class of Nonlinear Systems. In Proceedings of the IEEE Conference on Decision and Control (CDC). 2010.
2. J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. N. Kutz. On Dynamic Mode Decomposition: Theory and Applications. Journal of Computational Dynamics, 1(2):391–421. 2014.
3. Li, Weiwei, and Emanuel Todorov. Iterative Linear Quadratic Regulator Design for Nonlinear Biological Movement Systems. ICINCO (1). 2004.
4. Ansari, Alexander R., and Todd D. Murphey. Sequential Action Control: Closed-Form Optimal Control for Nonlinear and Nonsmooth Systems. IEEE Transactions on Robotics, 32(5):1196-1214. 2016.