Progress Report

Bardia Mojra

September 27, 2022

Robotic Vision Lab

The University of Texas at Arlington

1 Research Plan

This section outlines my current research plan where the main ideas, target conference/journal, and expected date of completion for each paper are provided. Target conferences: ICRA, IROS (March), CASE (Late Feb.), NIPS. Target Journals: RAL, CVPR, CORAL.

- Koopman-01 (IROS - Dec. 1st - active): Koopman-based MPC control of VTOL-DIP and VTOL-TIP in simulation, DLO pose estimation in simulation, experiments on choice of basis function and lifting dimensions, and performance comparison with optimal, robust, and/or adaptive control schemes.

- Koopman-02 (ACC - Sep 30th - active): A review on Koopman-based control schemes. Not enough, make it part of another paper. Read papers and write literature reviews.

- Koopman-03 (RAL - Mar. 1st - status): Extension to Koopman-01, Koopman-based dynamic estimation of DLO, collect dynamic DLO dataset, prediction of DLO configuration.

- Quest-01 (IROS - Mar. 1st - next): Optimal transform solution for QuEst based on dominant mode decomposition (DMD).

- Quest-02 (IROS/RAL - date - status): QuEst-based EKF, structure from motion, and VSLAM, compare performance with existing methods.

- Koopman-04 (IROS/RAL - date - status): Physics Informed (PI) Koopman-based control of a DLO, show obtained is persistant, compare to other non-PI methods, offline-online learning.

- Koopman-05 (IROS/RAL - date - status): PI Koopman operator (PIKO) based persistant model for DLOs, low dimensional, compare performance, offline-online learing/adapting, fast transfer learning.

- Koopman-06 (IROS/RAL - date - status): PIKO-based unit segment model for DLOs, more generalized, should yield better performance if number segments are selected online in order to obtain optimal representation in real-time given available hardware, compare results.

- Koopman-07 (IROS/RAL - date - status): DLO dataset, PIKO-based reinforcement learning of real DLO dynamics in a digital twin (DT) setting, experiments of model persistance, compare learning rate with neural network based methods, compare performance with available methods, and experiments on learing limitations.

- Koopman-08 (IROS/RAL - date - status): Koopman-based real-time control of DLO on GPU.
- Koopman-09 (IROS/RAL - date - status): PIKO-based real-time control of DLO on GPU.
- Koopman-10 (IROS/RAL - date - status): PIKO-based real-time control of deformable planar objects (DPO).
- Koopman-11 (IROS/RAL - date - status): PIKO-based real-time control of deformable volume objects (DVO).
- Koopman-12 (IROS/RAL - date - status): PIKO-based unit segment for DPOs, on GPU.
- Koopman-13 (IROS/RAL - date - status): PIKO-based unit segment for DVOs, on GPU.

2 To Do

- QEKF Paper (On pause):
  - Noise issue: noise cannot be modeled - DMD is a robust noise on high dimensional orthonormal time series and should be able to denoise QuEst solutions.
  - SfM: RQuEst cannot find solution - A potential solution is described briefly above.
- DLO Manipulation: (ICRA - section out of date)
  - Setup digital twin reinforcement learing setup:
    * Unity Robotics extension setup – done.
    * Design dynamic DLO data collection system.
    * Build work cell. – done
    * Collect data and create a dataset.
    * Define evaluation metrics.

    * Create a high frequency RGBD dataset with UV-frames and open-loop input control actions as the ground truth.

- Real-Time Preception – on hold
- Learning DLO Dynamics and System Identification - PIKO - Ongoing

3 Progress

The following items are listed in the order of priority:

- DLO Manipulation (IROS): Last week, I was finally able to understand how Koopman operator is implemented. I was able to find the implementation by Ian Abraham for their DLO paper [1]. I ported the code in from Python to Matlab and successfully run the project with a Van Der Pol model. As they explain, a Van Der Pol model is created for generating multiple runs with known random initial conditions and control input. Total of 10 runs with 200 time samples each are collected and concatenated. Multiple runs with random initial conditions is crucial for properly capturing dominant dynamics of the entire system. Moreover, the 2 dimensional system was lifted to 4 and a unique choice of basis function was used. The implementation is fully object oriented which, allows quick and easy reuse with any simulation or choice of basis function. Now, we want to replace Van Der Pol oscillator with a double-pendulum, experiment with different choices of basis function, and model predictive control (MPC) control schemes. Later in the week, I learned Matlab Simscape multi-body simulator and created a double-pendulum model. At this time, I am learning about Simscape models and objects and how they could be integrated into Matlab source code. This is a huge feat because the model can be extended without having to change the source code and Matlab code generator can be used to generate C, C++, and CUDA source code. This is where object oriented implementation makes all the difference and due to my implementation, the integration will require minimal effort.

- Maicol (REU): No update, he is busy with classes.

- DoD SMART (Dec 1st.): I started the application.

- XEst (RAL —): No update.

References

[1] I. Abraham, G. De La Torre, and T. D. Murphey, "Model-based control using koopman operators," *arXiv preprint arXiv:1709.01568*, 2017.