Thoughts about Generative Models in Control Systems

Prof. Raghvendra V. Cowlagi

Aerospace Engineering Department,
Worcester Polytechnic Institute, Worcester, MA.

rvcowlagi@wpi.edu **wpi.edu/∼rvcowlagi** **HL 247**

July 19, 2024

Introduction

- Control and estimation methods usually start with a dynamical system model of the form

$$x(k+1) = f(x(k), u(k), w(k)),$$

  where
  - x is what we call the *state*
  - u is what we call the *control*
  - w is what we call the *process noise*

- Usual assumptions include:
  - $x(k) \in \mathbb{R}^n, \; u(k) \in \mathbb{R}^m$, where n, m are "small" ~ 10
  - f is known, e.g., from physics
  - w is a white noise process, i.e., $w(k) \sim N(0, Q)$ and $w(k), w(\ell)$ are uncorrelated

Nominal Model

- Control and estimation methods usually start with a dynamical system model of the form

$$x(k+1) = f(x(k), u(k), w(k)),$$

  where
  - x is what we call the state
  - u is what we call the control
  - w is what we call the process noise

- Let us call this the *nominal model* (to distinguish from other "models" that come up)

Nominal v/s Real

- The real system evolves *close to,* but not *exactly* according to the nominal model. Examples:
- f is slightly different, e.g., has some extra terms due to simplifications, approximations, epistemic uncertainties
- w is not white noise
- There are extra (hidden) states, e.g., actuator dynamics

The Uncertainty Propagation Problem

- We are usually interested in the conditional distribution $p(x(k+1) \mid x(k), u(k))$ for designing estimators or controllers (esp. RL-based controllers)
  - Estimators like EKF / UKF need the mean and variance
  - Particle filter, RL methods need samples
- *Option 1* (most common practice): Find this distribution from the nominal model
  - This approach involves approximations when f is nonlinear: e.g., linearization or unscented transform
  - Essentially, we entirely neglect the differences between reality and nominal model and hope that the controller / estimator is robust enough
- *Option 2* (the adaptive control approach): Append a "structured uncertainty" term Φ to f and learn its parameters through open- and closed-loop training
  - Φ often resembles a single-layer NN, e.g., weighted sum of RBFs
  - Sometimes used in estimation as well

Data

- We have some *real trajectories*, i.e., time series $\tilde{\xi}(k) = (\tilde{x}(k), \tilde{u}(k))$ recorded during operation of the real system
  - These are scarce
- We have some *nominal trajectories*, time series $\xi(k) = (x(k), u(k))$ resulting from simulation of the nominal model
  - These are abundant

Rationale

- The unscented transform is motivated by the idea that it is better to approximate the *distribution* propagated through f, than to approximate f itself (e.g., by linearization)
- Which suggests that Options 1 and 2 may be outperformed by ...
- *Option 3:* Train a conditioned generative model on both real and nominal trajectories to produce samples of $x(k+1)$ for given $x(k), u(k)$

Case Study 1

- Nominal model is linear:

$$x(k+1) = Ax(k) + Bu(k)$$

- Suppose $x(k) \in \mathbb{R}^2,\ u(k) \in \mathbb{R}$ and A is Hurwitz: $Re(\text{both eigenvalues}) < 0$
- Real system does not have process noise, but has unmodeled terms:

$$x(k+1) = Ax(k) + Bu(k) + \tilde{f}(x(k))$$

Questions for Randy

- Are there generative models in other domains (e.g., text / image) that address this type of a problem?

- Perhaps a "variational propagator"?

$$\xi = (x, u) \rightarrow p(z|\xi) \qquad z \sim p(z|\xi) \rightarrow D(z)$$

  - Loss= $\lambda_1 \|D(z) - x(k+1)\| + \lambda_2 KL(p\,\|\mathcal{N}(0, I)) + \lambda_3 \underbrace{\|x - f(x,u)\|}_{\text{or } |\|x - f(x,u)\| - \varepsilon|}$

- How can we "condition" this on $x(k), u(k)$?