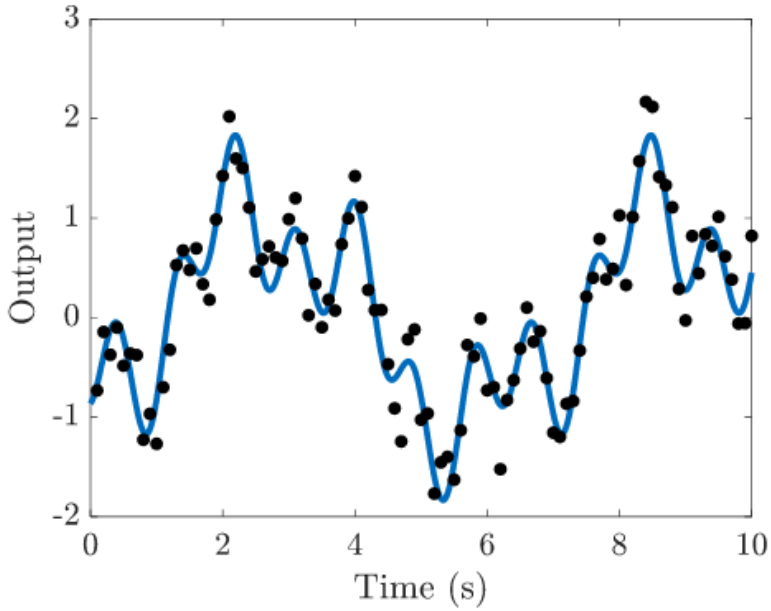
Bayesian Approaches for Data-Driven Learning of Dynamical Systems

Engineering Mechanics Institute Conference and Probabilistic Mechanics & Reliability Conference (Virtual)
May 25-28, 2021

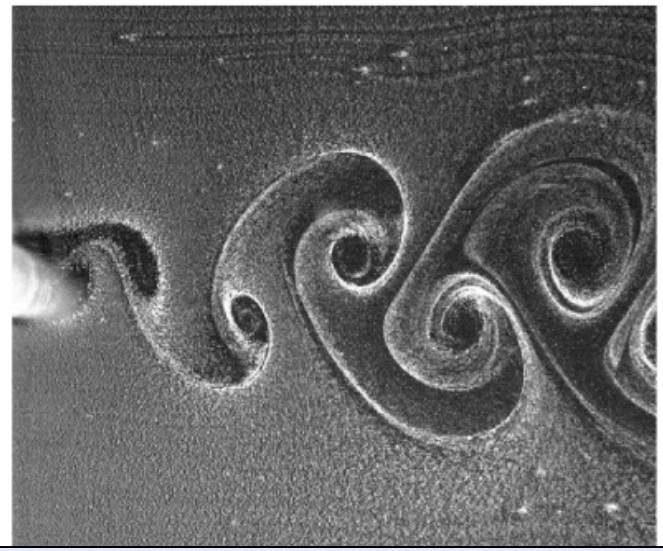
Alex Gorodetsky and Nicholas Galioto
Aerospace Engineering, UM

UNIVERSITY OF MICHIGAN

Motivation

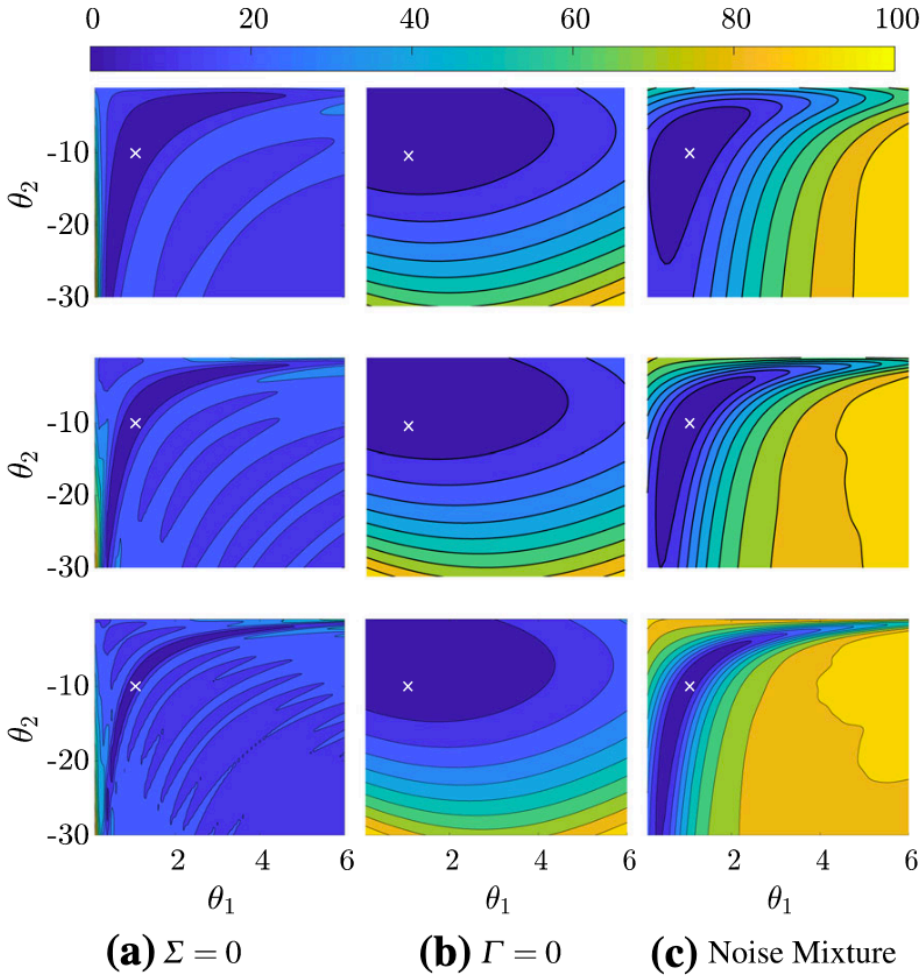


- We seek to learn physically relevant models from time-series data in a way that
  - Handles sparse and noisy data
  - Scales well with dimension
  - Captures uncertainty in predictions
- Two primary design choices must be made
  - Model structure: linear, linear subspace, neural networks, etc.
  - Objective functions: least squares, regularization, Bayesian posterior



Main takeaway

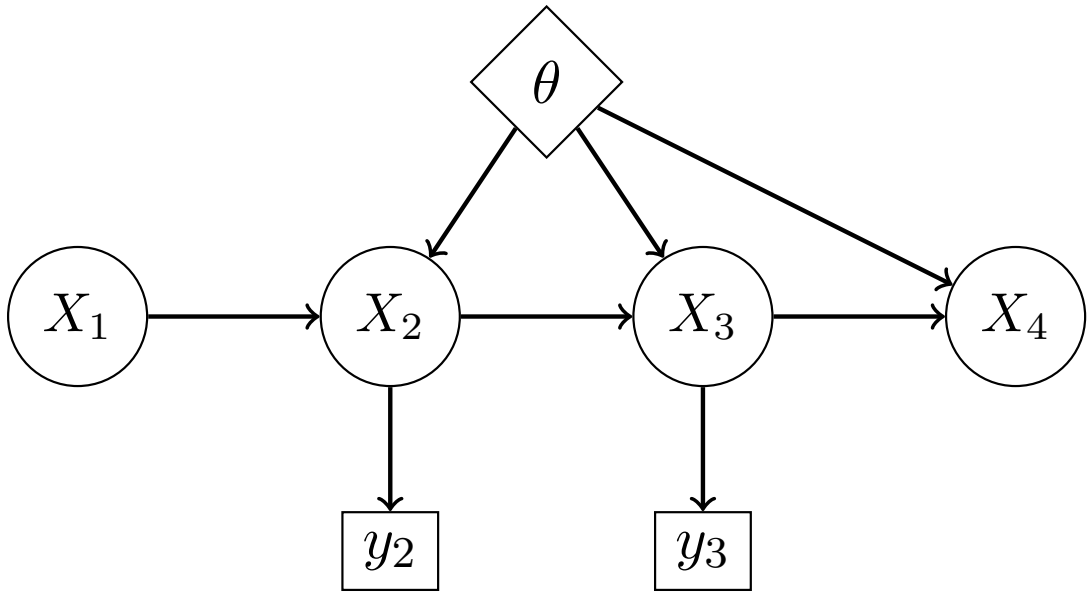
- A majority of existing approaches minimize least-squares objectives
  1. Assume perfect model
     $\dot{x} = f(t, x; \theta)$
     $J(\theta) = \sum_{i=1}^{n} \|y_i - x(t_i)\|_2^2$
  2. Effectively assume noiseless measurements (DMD)
     $J(\theta) = \sum_{i=1}^{n} \|y_i - \Psi(y_{i-1}; \theta)\|_2^2$
- Our approach: assume noisy measurements + ***stochastic model***
  - Provides optimal combination of (1) and (2)
  - Provides natural dynamics-based regularization



Parameterized Hidden Markov Model

Generic formulation: linear, physical model, dictionary of bases, NN, etc.

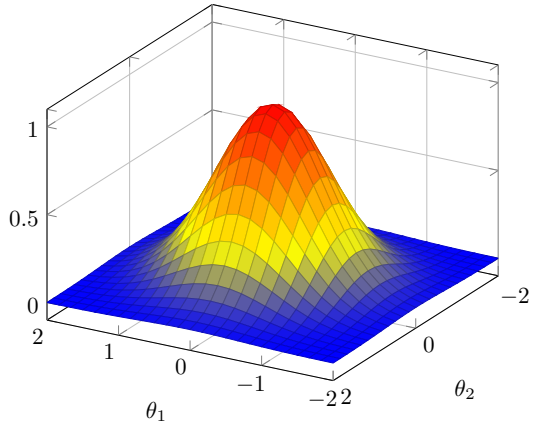
$$X_{k+1} = \Psi(X_k; \theta) + \eta_k$$
$$Y_k = h(X_k) + \nu_k,$$
$$\eta_k \sim \mathcal{N}(0, \Sigma)$$
$$\nu_k \sim \mathcal{N}(0, \Gamma)$$



1. Accounts for model parameter uncertainty through θ
2. Accounts for model form uncertainty through η_k
3. Accounts for measurement noise through ν_k

Bayesian inference
Updating models with data

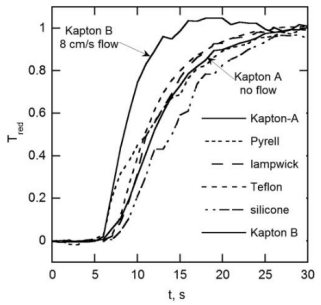
Prior $p(\theta, \mathcal{X}_n | I)$



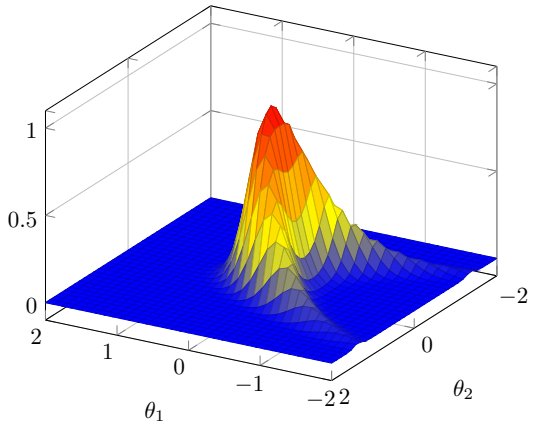
Bayes' Rule

$$p(\theta, \mathcal{X}_n | \mathcal{Y}_n, I) = p(\theta, \mathcal{X}_n \mid I) \frac{p(\mathcal{Y}_n | \theta, \mathcal{X}_n, I)}{p(\mathcal{Y}_n | I)}$$

Data



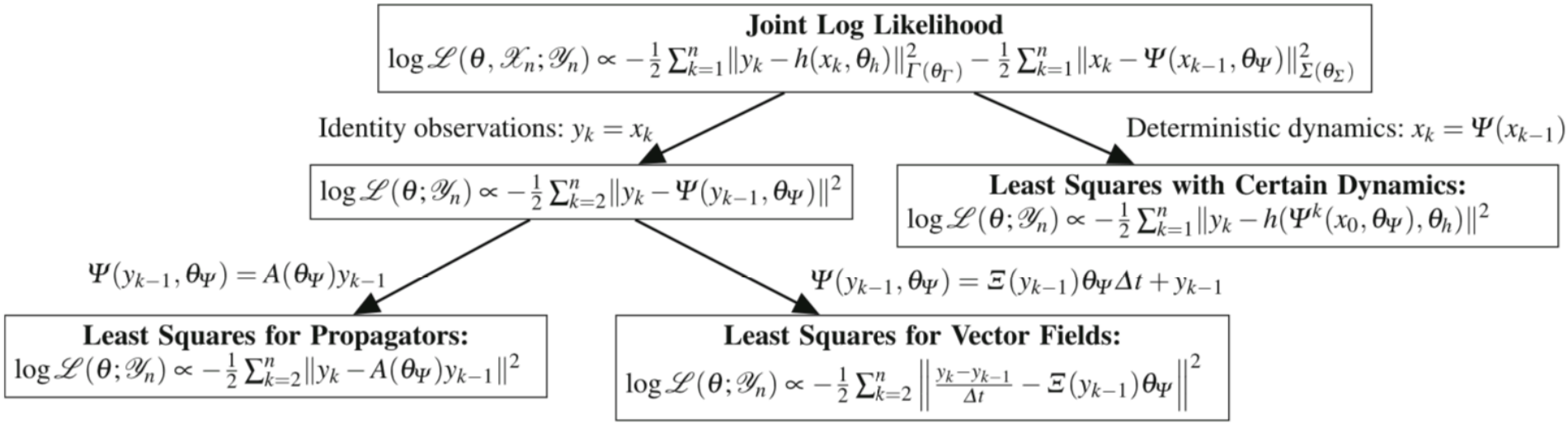
Posterior $p(\theta, \mathcal{X}_n | \mathcal{Y}_n, I)$



For our Markovian system

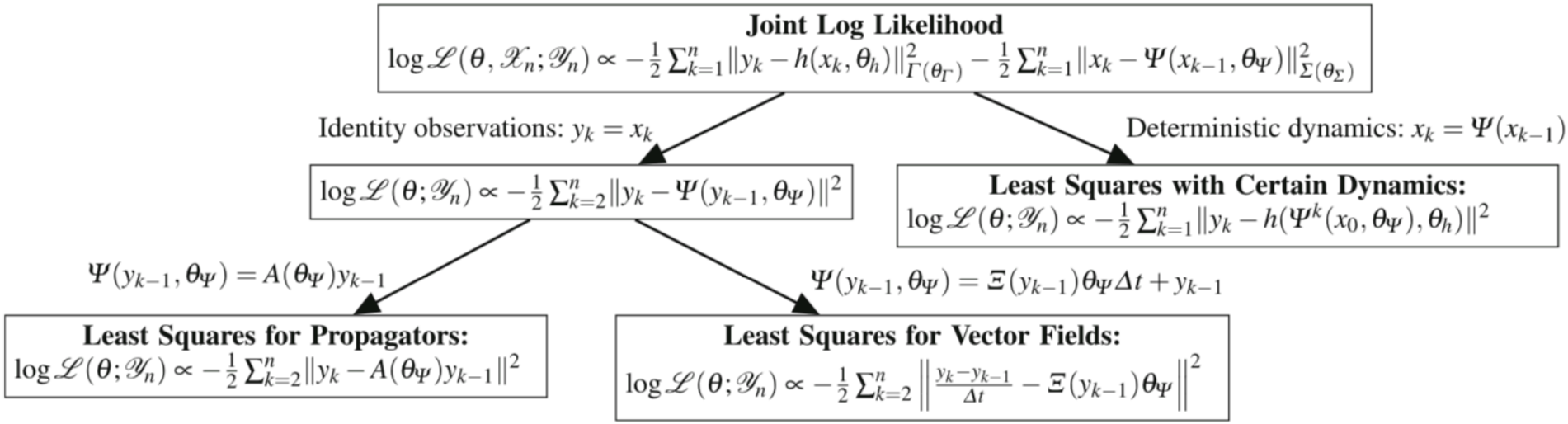
$$\underbrace{p(\theta, \mathcal{X}_n \mid \mathcal{Y}_n, I)}_{\text{posterior}} = \frac{1}{Z} \underbrace{\left[\prod_{i=1}^{n} p(y_i \mid X_i, \theta)\right]}_{\text{likelihood}} \underbrace{\left[\prod_{i=1}^{n} p(X_i \mid X_{i-1}, \theta)\right] p(\theta \mid I)}_{\text{prior}}$$

Log posterior is a generalized objective function
Many existing approaches recovered under simplifying assumptions



1. Identity observations: Hills et. al. 2015, Raissi, 2018, Qin et.al 2019
2. Deterministic dynamics: Ayed et. al. 2019, Long et. al. 2018, Zhong et. al. 2019
3. Many of these works look at parameterizing models (e.g., via NN architectures)

Log posterior is a generalized objective function
Many existing approaches recovered under simplifying assumptions



1. Identity observations: Hills et. al. 2015, Raissi, 2018, Qin et.al 2019
2. Deterministic dynamics: Ayed et. al. 2019, Long et. al. 2018, Zhong et. al. 2019
3. Many of these works look at parameterizing models (e.g., via NN architectures)

Ayed, I., de Bézenac, E., Pajot, A., Brajard, J., & Gallinari, P. (2019). Learning dynamical systems from partial observations. arXiv preprint arXiv:1902.11136.; Long, Z., Lu, Y., Ma, X., & Dong, B. (2018, July). Pde-net: Learning pdes from data. In International Conference on Machine Learning (pp. 3208-3216).; Zhong, Y. D., Dey, B., & Chakraborty, A. (2019). Symplectic ode-net: Learning hamiltonian dynamics with control. arXiv preprint arXiv:1909.12077. ; Hills, D. J., Grütter, A. M., & Hudson, J. J. (2015). An algorithm for discovering Lagrangians automatically from data. PeerJ Computer Science, 1, e31. ; Qin, T., Wu, K., & Xiu, D. (2019). Data driven governing equations approximation using deep neural networks. Journal of Computational Physics, 395, 620-635. ; Raissi, M. (2018). Deep hidden physics models: Deep learning of nonlinear partial differential equations. The Journal of Machine Learning Research, 19(1), 932-955.

Goals: learn and predict

- Learn the model: $p(\theta \mid \mathcal{Y}_n, I)$
- Make (future) predictions : $p(X_k \mid \mathcal{Y}_n, I) = \int p(X_k \mid \theta) p(\theta \mid \mathcal{Y}_n, I) d\theta$

Marginal posterior — required by Optimization/Markov Chain Monte Carlo

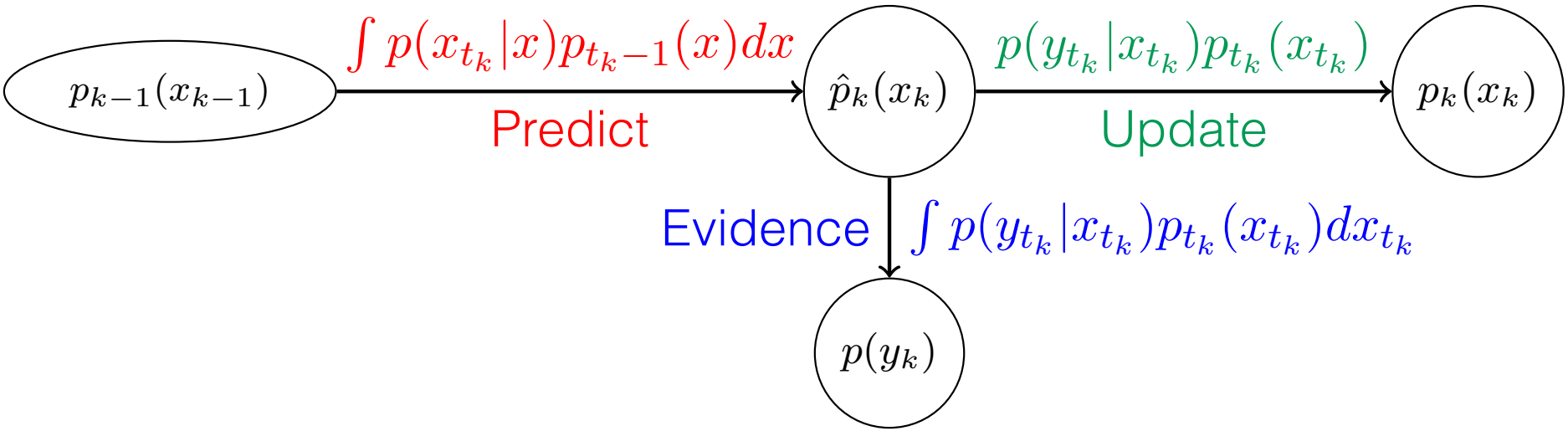
$$p(\theta \mid \mathcal{Y}_n, I) = \int p(\theta, \mathcal{X}_n \mid \mathcal{Y}_n, I) d\mathcal{X}_n = \frac{1}{Z} \underbrace{p(\mathcal{Y}_n \mid \theta)}_{\text{marginal likelihood}} p(\theta)$$

Andrieu, Christophe, and Gareth O. Roberts. "The Pseudo-Marginal Approach for Efficient Monte Carlo Computations." The Annals of Statistics 37.2 (2009): 697–725.

Evaluating the marginal posterior

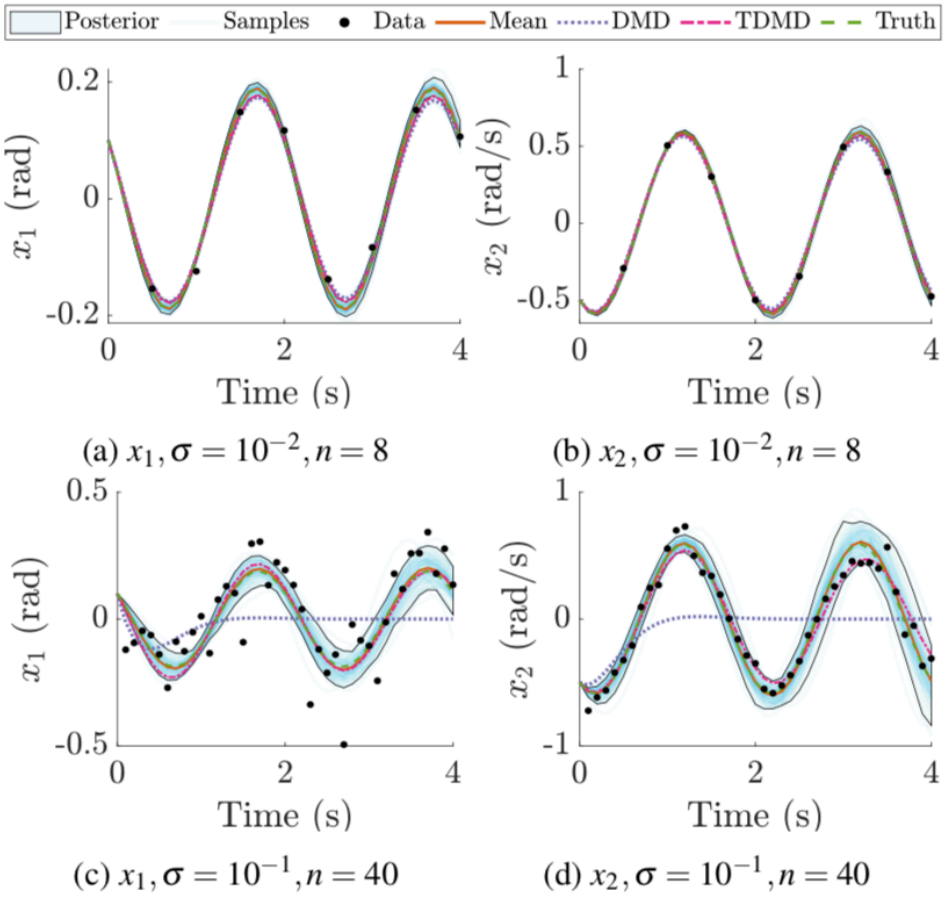
Recursive evaluation $p(\theta \mid \mathcal{Y}_n)$

1: **for** $k = 1$ to n **do**
2: Predict $p(X_k|\theta, \mathcal{Y}_{k-1}) = \int p(X_k|\theta, X_{k-1})p(X_{k-1}|\theta, \mathcal{Y}_{k-1})dX_{k-1}$
3: Compute the evidence $p(y_k|\theta, \mathcal{Y}_{k-1}) = \int p(y_k|\theta, X_k)p(X_k|\theta, \mathcal{Y}_{k-1})dX_k$
4: Update filter $p(X_k|\theta, \mathcal{Y}_k) = \frac{p(y_k|\theta, X_k)p(X_k|\theta, \mathcal{Y}_{k-1})}{p(y_k|\theta, \mathcal{Y}_{k-1})}$
5: Update posterior $p(\theta|\mathcal{Y}_k) = \frac{p(y_k|\theta, \mathcal{Y}_{k-1})p(\theta|\mathcal{Y}_{k-1})}{p(y_k|\mathcal{Y}_{k-1})}$
6: **end for**



Linear pendulum: DMD model is "correct"
Correct DMD model still does not recover system

Reconstruction

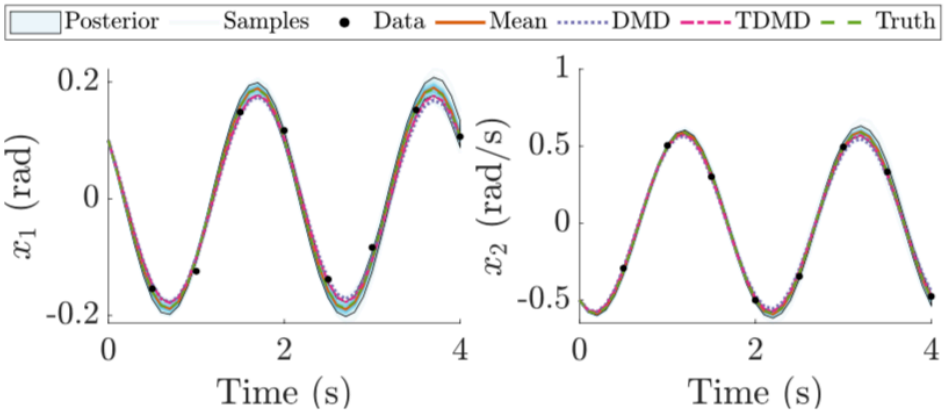


(a) $x_1, \sigma = 10^{-2}, n = 8$ (b) $x_2, \sigma = 10^{-2}, n = 8$

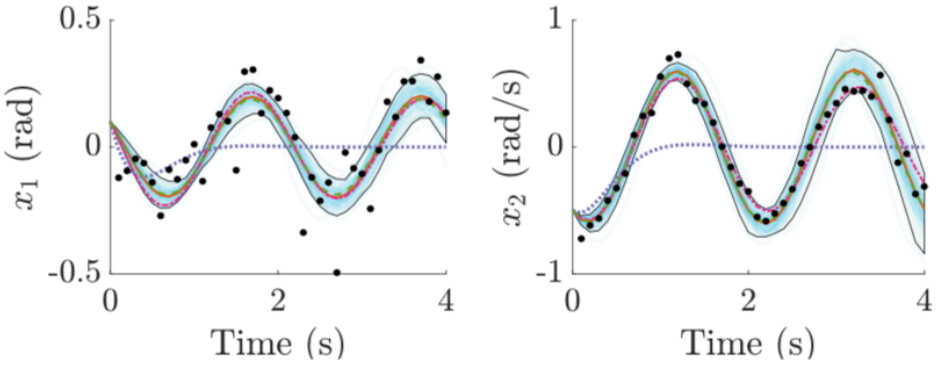
(c) $x_1, \sigma = 10^{-1}, n = 40$ (d) $x_2, \sigma = 10^{-1}, n = 40$

Linear pendulum: DMD model is "correct"
Correct DMD model still does not recover system

Reconstruction

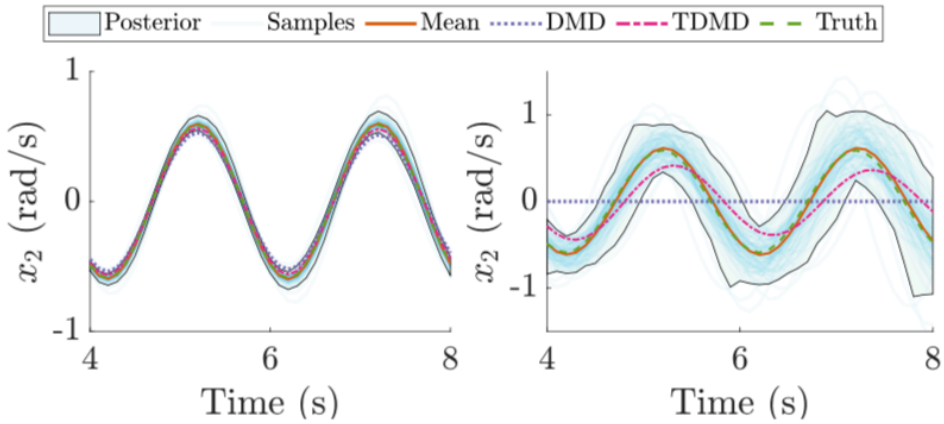


(a) $x_1, \sigma = 10^{-2}, n = 8$ (b) $x_2, \sigma = 10^{-2}, n = 8$



(c) $x_1, \sigma = 10^{-1}, n = 40$ (d) $x_2, \sigma = 10^{-1}, n = 40$

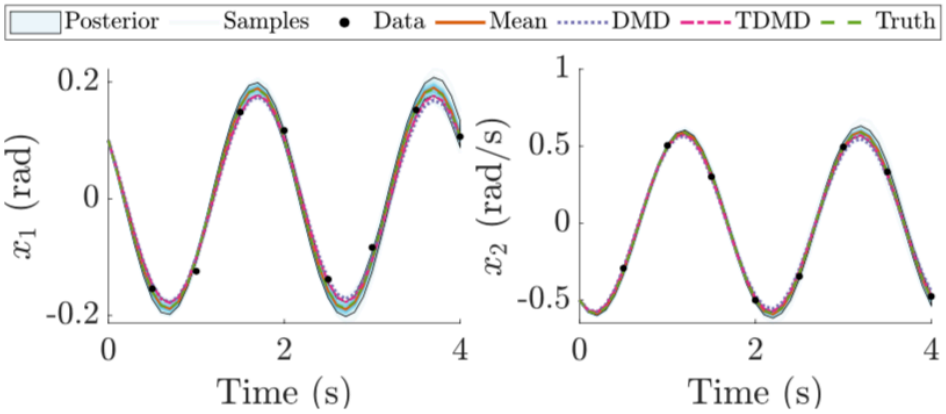
Prediction



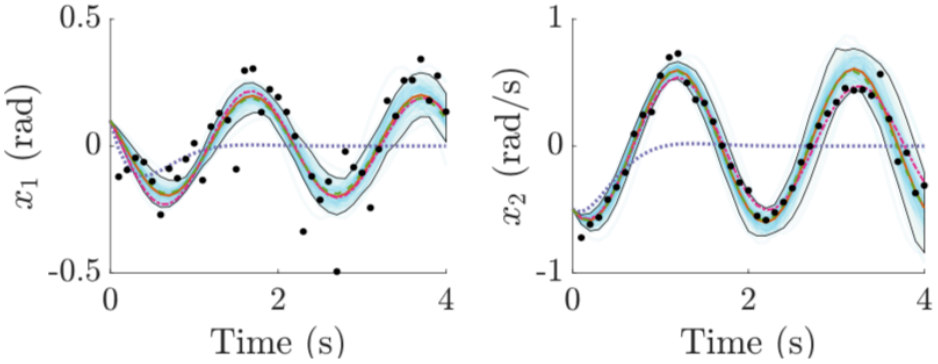
(a) $x_2, \sigma = 10^{-2}, n = 8$ (b) $x_2, \sigma = 10^{-1}, n = 40$

Linear pendulum: DMD model is "correct"
Correct DMD model still does not recover system

Reconstruction

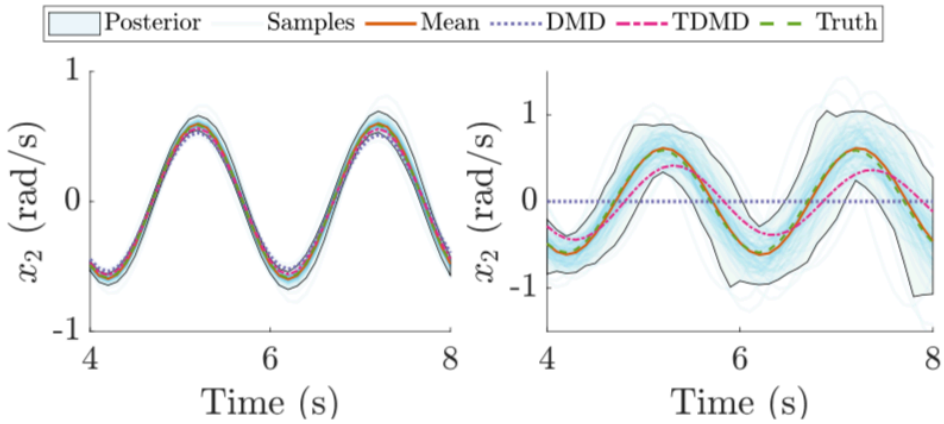


(a) $x_1, \sigma = 10^{-2}, n = 8$ (b) $x_2, \sigma = 10^{-2}, n = 8$



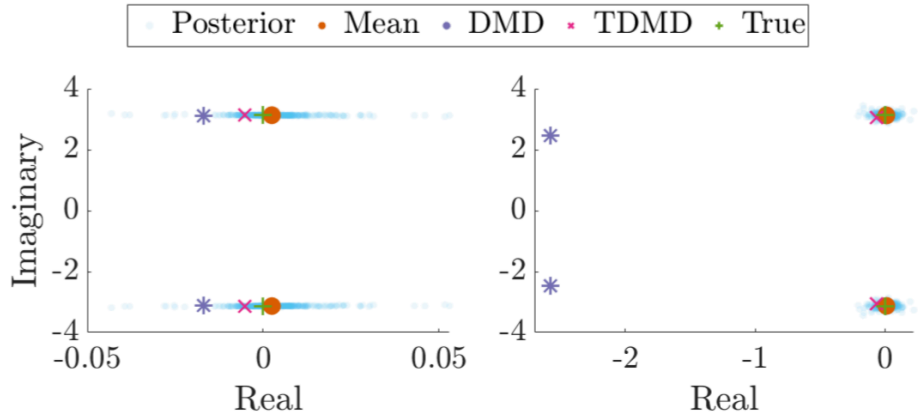
(c) $x_1, \sigma = 10^{-1}, n = 40$ (d) $x_2, \sigma = 10^{-1}, n = 40$

Prediction



(a) $x_2, \sigma = 10^{-2}, n = 8$ (b) $x_2, \sigma = 10^{-1}, n = 40$

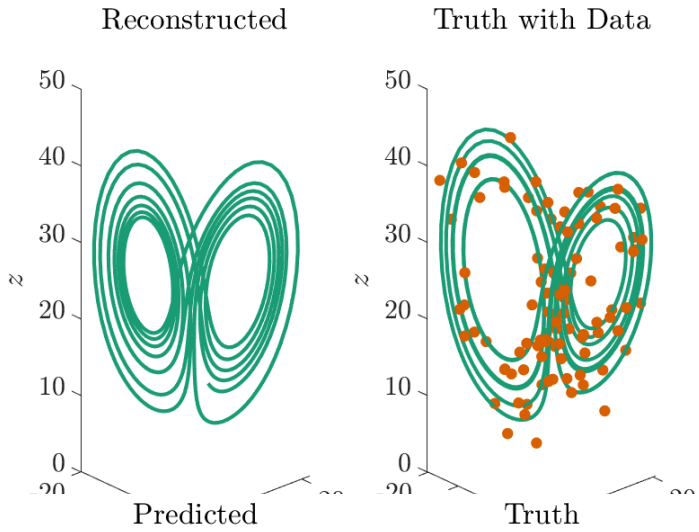
Spectrum



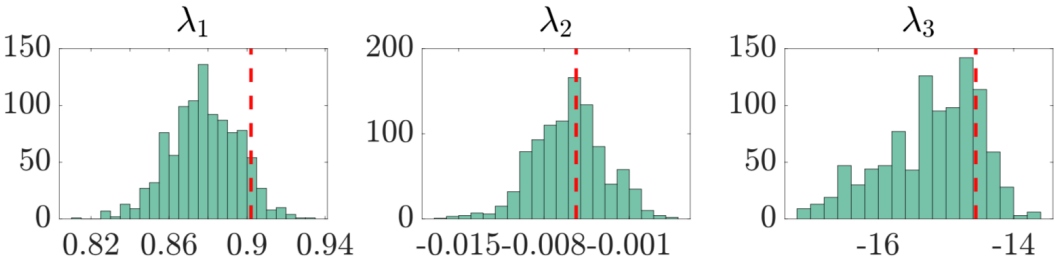
(a) Lower n, σ (b) Higher n, σ

Lorenz 63
With only 300 noisy data points, we recover the attractor

$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = x(\rho - z) - y$$
$$\dot{z} = xy - \beta z$$



Posterior Lyapuynov exp.



(a) λ_1 Estimate (b) λ_2 Estimate (c) λ_3 Estimate

Specializing to Hamiltonian systems

- Hamiltonian systems are reversible and preserve certain invariants (energy)

$$\dot{q} = \frac{\partial H}{\partial p} \quad \dot{p} = -\frac{\partial H}{\partial q}$$

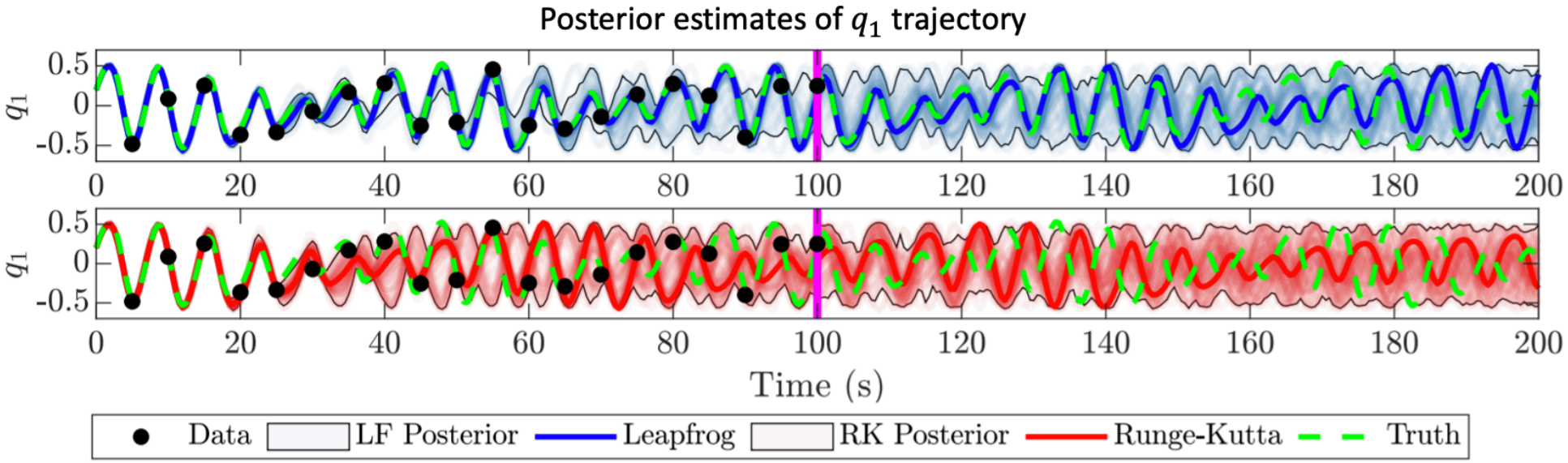
- Dynamics Ψ become a mixture of a leap frog and Hamiltonian parameterization: $H = \frac{1}{2} p^T M^{-1}(q) p + U(q,p)$

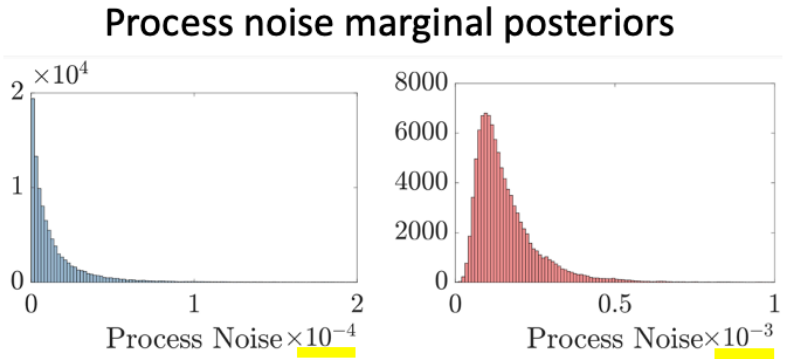
$$\Psi(q_k, p_k; \theta_\Psi) = \begin{bmatrix} q_k + \Delta t p_k - \frac{\Delta t^2}{2} \left.\frac{\partial U(q,p,\theta_\Psi)}{\partial q}\right|_{q_k} \\ p_k - \frac{\Delta t}{2}\left(\left.\frac{\partial U(q,p,\theta_\Psi)}{\partial q}\right|_{q_k} + \left.\frac{\partial U(q,p,\theta_\Psi)}{\partial q}\right|_{q_{k+1}} \right) \end{bmatrix},$$

- We parameterize the potential energy U
- We both learn the Hamiltonian and assume the data is from a symplective process

Hénon Heiles Potential

$$U(q_1, q_2) = \frac{1}{2}q_1^2 + \frac{1}{2}q_2^2 + q_1^2 q_2 - \frac{1}{3}q_2^3$$





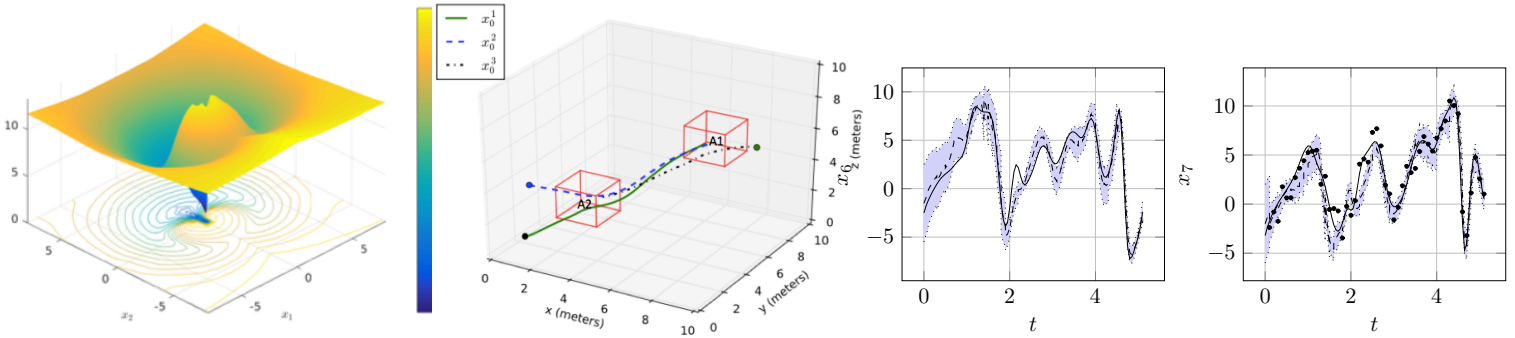
Symplectic approach learns a model with an order of magnitude greater certainty

Conclusions and Acknowledgements

1. Conclusions
   - Learning stochastic models is benefitial for system ID to handle
     - Parameter uncertainty
     - Model uncertainty
     - Measurement uncertainty
   - MCMC approaches are feasible for moderately large problems
   - How do we know the size of the latent space?
2. Papers on my website
   `www.alexgorodetsky.com/publications.html`
   1. Galioto, N., Gorodetsky, A. A. "Bayesian system ID: optimal management of parameter, model, and measurement uncertainty." Nonlinear Dyn (2020).
   2. Galioto, N. Gorodetsky, A. A., "Bayesian Identification of Hamiltonian Dynamics from Symplectic Data." Conference on Decision and Control (CDC), 2020
3. Funding: DARPA PAI and AIRA, AFOSR Computational Mathematics

Computational science for autonomy

Compression enabled control and estimation



Real-time autonomy

- Source code: `github.com/goroda`, papers: `alexgorodetsky.com`
- Thanks!