Bayesian Learning of Stochastic Dynamical Models for Quantities of Interest

Nicholas Galioto (ngalioto@umich.edu) and Alex Arkady Gorodetsky

Department of Aerospace Engineering, University of Michigan – Ann Arbor

SIAM Conference on Uncertainty Quantification

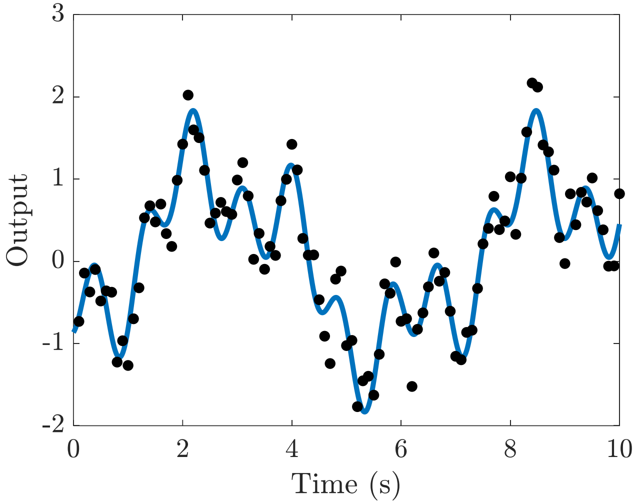
April 15, 2022

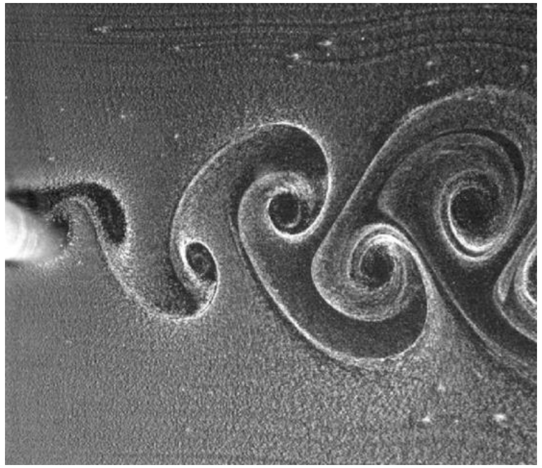
Motivation

Objective: learn a model of a dynamical system from data

Two primary design choices in system identification:

- Model structure
  - Neural networks
  - Basis expansions
  - Kernels
- **Objective function** (our interest)
  - Least squared error
  - Regularization





A good objective will:

- Be robust to sparse and noisy data
- Handle model inadequacy
- Trade off bias and variance optimally

ngalioto@umich.edu
Ljung, L. (1999). System identification. *Wiley encyclopedia of electrical and electronics engineering*, 1-19.
Schoukens, J., & Ljung, L. (2019). Nonlinear system identification: A user-oriented road map. *IEEE Control Systems Magazine*, *39*(6), 28-99.

Partially Observed Systems

Identifying models from partial observations is challenging

- Latent space/coordinate frame is unknown
- Incomplete information on the full state
- Cannot use basis functions that depend on the full state

GPS Satellite ©Lockheed Martin

Contributions

Present an algorithm that can:

- Handle measurement, model, and parameter uncertainty and their interaction
- Quantify model uncertainty
- Learn models of chaotic systems from partial observations
- Model the dynamics of a PDE quantity of interest without modeling the full field

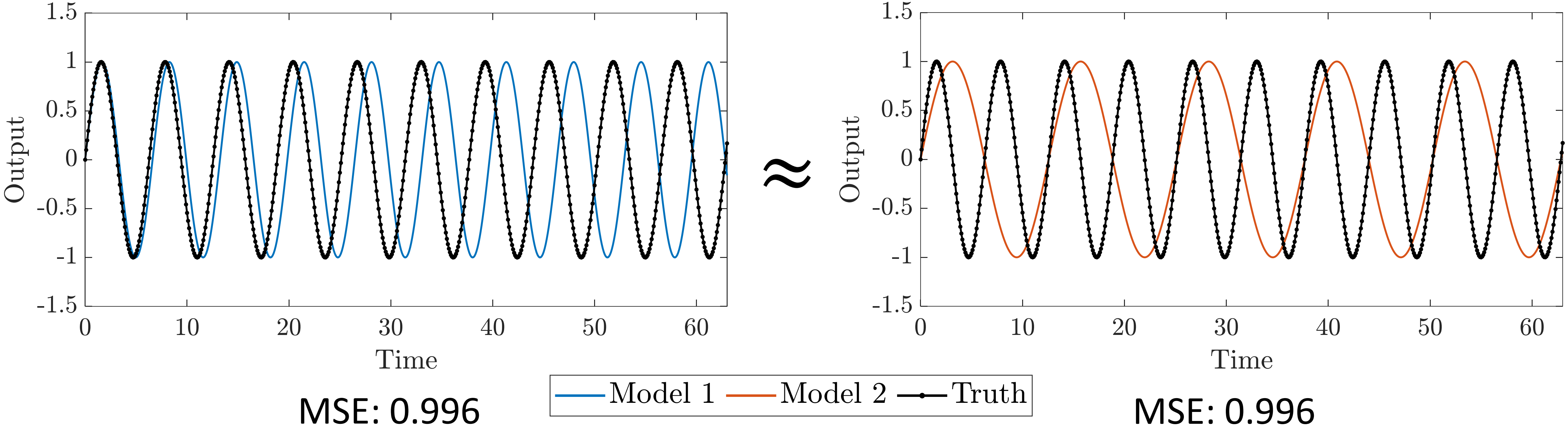
ngalioto@umich.edu

Outline

1. Existing approaches
2. Probabilistic formulation
3. Algorithm/Marginal likelihood
4. Results
5. Takeaways

The mean squared error metric can induce an undesirable ranking of dynamical models

The accumulation of small model errors is given equal weight as large model error



MSE: 0.996

MSE: 0.996

How can we design an objective that prioritizes Model 1 over Model 2?

Existing Approaches

Least squares-based objective functions

(a) Assumes perfect model

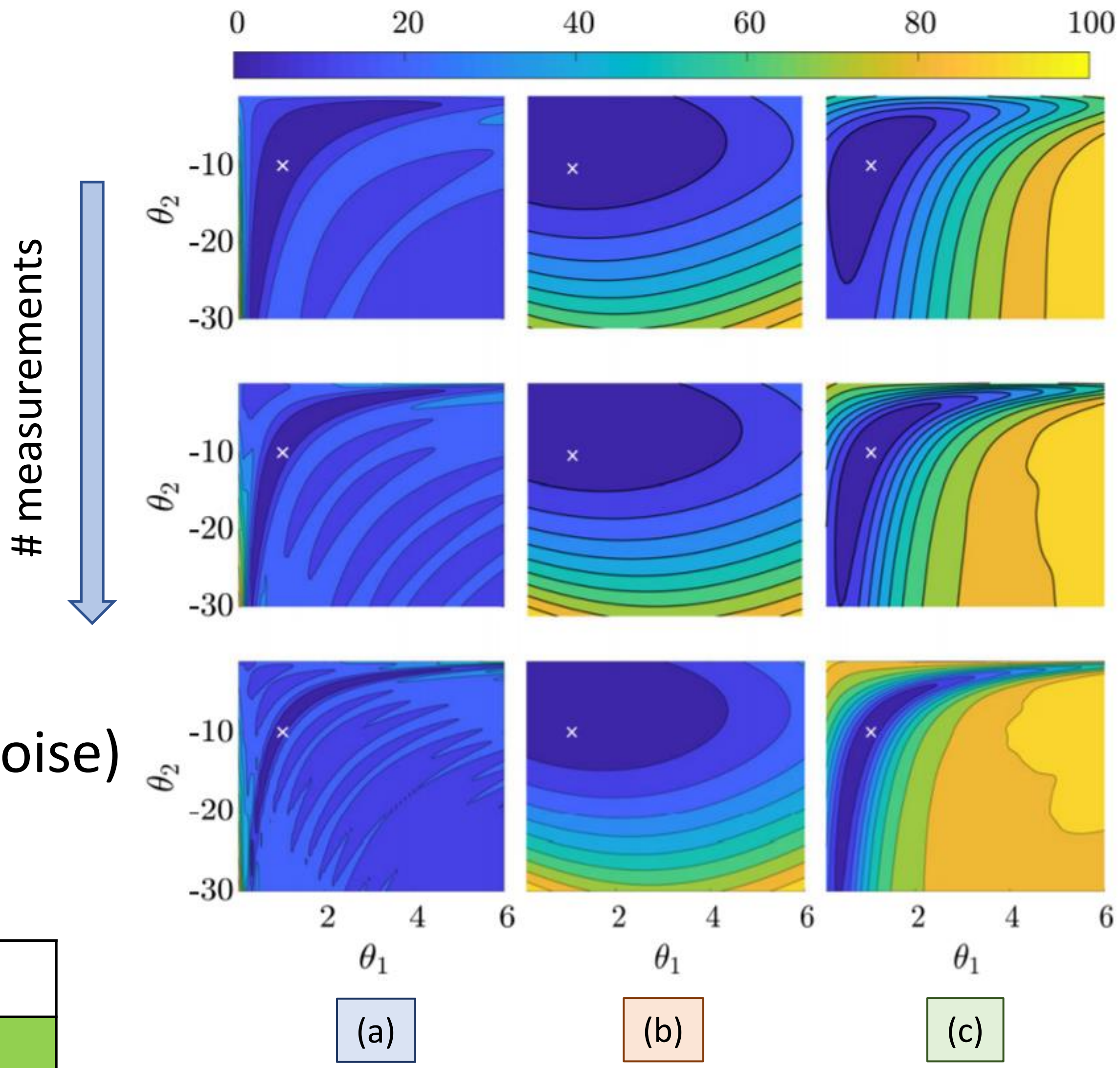
$$J(\theta) = \sum_{k=1}^{n} \|y_k - h(x(t_k), \theta)\|_2^2 \quad \text{s.t.} \quad \frac{\mathrm{d}x}{\mathrm{d}t} = f(t, x; \theta)$$

(b) Assumes noiseless measurements

$$J(\theta) = \sum_{k=1}^{n} \|y_k - \Psi(y_{k-1}; \theta)\|_2^2$$

(c) Noisy measurements + model error (process noise)

- Optimal combination of (a) and (b)



	(a)	(b)	(c)
Steep optimization surfaces without plateaus	✓	×	✓
Smooths local minima	×	✓	✓
Increased confidence with data	✓	×	✓

Outline

1. Existing approaches
2. Probabilistic formulation
3. Algorithm/Marginal likelihood
4. Results
5. Takeaways

Probabilistic Formulation

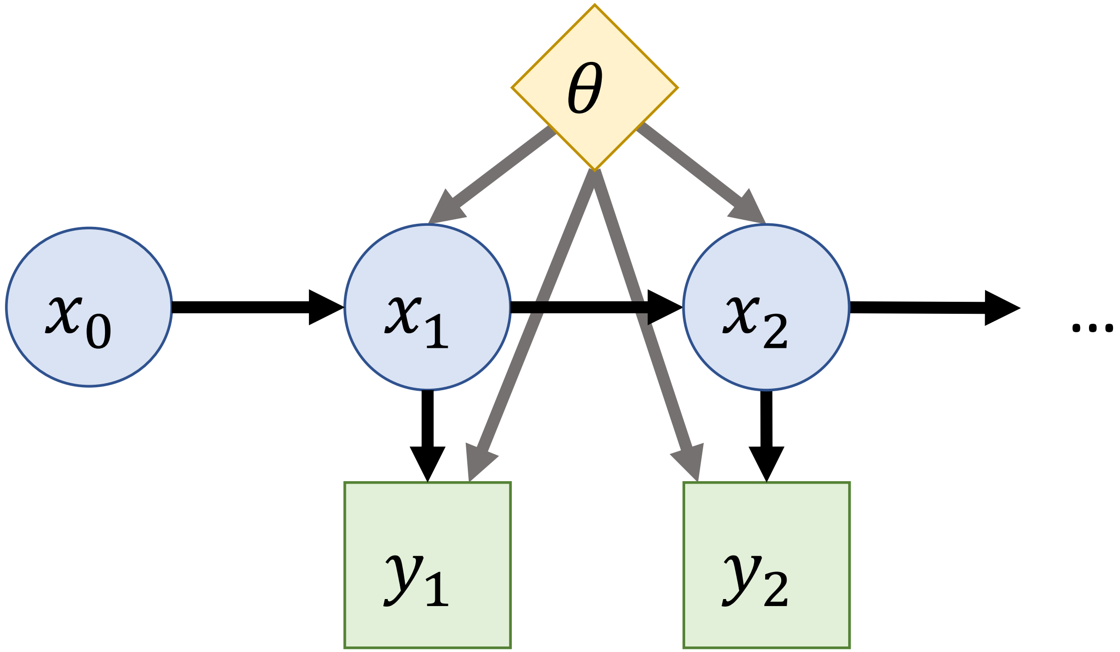
Joint parameter-state estimation with stochastic dynamics

$$X_k \in \mathbb{R}^{d_x}, \qquad Y_k \in \mathbb{R}^{d_y}, \qquad \theta = (\theta_\Psi, \theta_h, \theta_\Sigma, \theta_\Gamma) \in \mathbb{R}^{d_\theta}$$

$$X_k = \Psi(X_{k-1}, u_{k-1}, \theta_\Psi) + \xi_k; \quad \xi_k \sim \mathcal{N}\big(0, \Sigma(\theta_\Sigma)\big)$$

$$Y_k = h(X_k, \theta_h) + \eta_k; \qquad \eta_k \sim \mathcal{N}\big(0, \Gamma(\theta_\Gamma)\big)$$

The process noise term ξ_k accounts for model error
- Parameter error
- Integration error
- Insufficient model expressiveness



1. Parameter Uncertainty
2. Model Uncertainty
3. Measurement Uncertainty

Posterior Flow Chart

Log Joint Likelihood

$$\log \mathcal{L}(\theta;\ \mathcal{X}_n, \mathcal{Y}_n) \propto -\frac{1}{2}\sum_{k=1}^{n}\|y_k - h(x_k, \theta_h)\|^2_{\Gamma(\theta_\Gamma)} - \frac{1}{2}\sum_{k=1}^{n}\|x_k - \Psi(x_{k-1}, \theta_\Psi)\|^2_{\Sigma(\theta_\Sigma)}$$

Deterministic dynamics:
$x_k = \Psi(x_{k-1})$

$$\log \mathcal{L}(\theta;\ \mathcal{Y}_n) \propto -\frac{1}{2}\sum_{k=1}^{n}\left\|y_k - h\left(\Psi^k(x_0, \theta_\Psi), \theta_h\right)\right\|^2$$

- ODE-Net; Chen et al., 2018
- PDE-Net; Long et al., 2018
- UDE; Rackauckas et al., 2019

Identity observations:
$y_k = x_k$

$$\log \mathcal{L}(\theta;\ \mathcal{Y}_n) \propto -\frac{1}{2}\sum_{k=2}^{n}\|y_k - \Psi(y_{k-1}, \theta_\Psi)\|^2$$

- DMD; Schmid, 2010
- SINDy; Brunton et al., 2019
- Hamiltonian NN; Greydanus et al., 2019

ngalioto@umich.edu

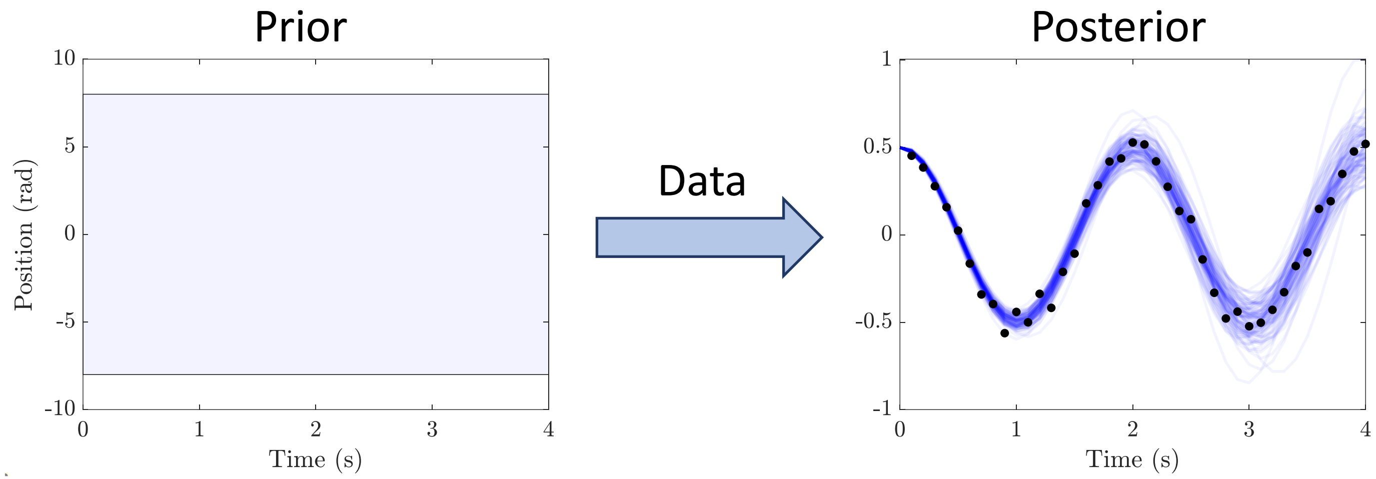
Chen, R. T., Rubanova, Y., Bettencourt, J., & Duvenaud, D. K. (2018). Neural ordinary differential equations. *Advances in neural information processing systems*, *31*.
Long, Z., Lu, Y., Ma, X., & Dong, B. (2018, July). Pde-net: Learning pdes from data. In *International Conference on Machine Learning* (pp. 3208-3216).
Rackauckas, C., Ma, Y., Martensen, J., Warner, C., Zubov, K., Supekar, R., ... & Edelman, A. (2020). Universal differential equations for scientific machine learning. *arXiv preprint arXiv:2001.04385*.
Schmid, P. J. (2010). Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics*, *656*, 5-28.
Brunton, S. L., Proctor, J. L., & Kutz, J. N. (2016). Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the national academy of sciences*, *113*(15), 3932-3937.
Greydanus, S., Dzamba, M., & Yosinski, J. (2019). Hamiltonian neural networks. *Advances in Neural Information Processing Systems*, *32*.

Outline

1. Existing approaches
2. Probabilistic formulation
3. Algorithm/Marginal likelihood
4. Results
5. Takeaways

Bayesian Inference

- Goal: compute $p(\theta|\mathcal{Y}_n)$ where $\mathcal{Y}_n = (y_1, y_2, \dots, y_n)$
- Bayes' rule: $p(\theta|\mathcal{Y}_n) = \dfrac{\mathcal{L}(\theta; \mathcal{Y}_n)p(\theta)}{p(\mathcal{Y}_n)}$



- Due to uncertainty in the states, we can only access the joint likelihood: $\mathcal{L}(\theta;\ \mathcal{X}_n, \mathcal{Y}_n)$
- To get the marginal likelihood, we must evaluate the integral

$$\mathcal{L}(\theta;\ \mathcal{Y}_n) = \int \mathcal{L}(\theta;\ \mathcal{X}_n, \mathcal{Y}_n) d\mathcal{X}_n$$

Marginal Markov Chain Monte Carlo (MCMC) Algorithm (Särkkä, 2013)

1. **for** $i = 1, \dots, N$
2. Propose sample θ

   Evaluate posterior: $p(\theta|\mathcal{Y}_n) = p(\theta)\prod_{k=1}^{n}\mathcal{L}_k(\theta;\mathcal{Y}_k)$
3. **for** $k = 0, \dots, n-1$
4. Predict: $p(X_{k+1}|\mathcal{Y}_k,\theta) = \int p(X_{k+1}|X_k,\theta)p(X_k|\mathcal{Y}_k,\theta)dX_k$
5. Marginalize: $\mathcal{L}_{k+1}(\theta;\mathcal{Y}_{k+1}) = \int p(y_{k+1}|X_{k+1},\theta)p(X_{k+1}|\mathcal{Y}_k,\theta)dX_{k+1}$
6. Update: $p(X_{k+1}|\mathcal{Y}_{k+1},\theta) = \frac{p(y_{k+1}|X_{k+1},\theta)p(X_{k+1}|\mathcal{Y}_k,\theta)}{p(y_{k+1}|\mathcal{Y}_k,\theta)}$
7. **end for**
8. Accept θ with Metropolis-Hastings probability; otherwise reject
9. **end for**

Kalman Filter / Probabilistic Filter (steps 4–6)

MCMC (steps 1–9)

ngalioto@umich.edu
Särkkä, S. (2013). *Bayesian filtering and smoothing* (No. 3). Cambridge University Press.

Recursive Marginal Likelihood Evaluation

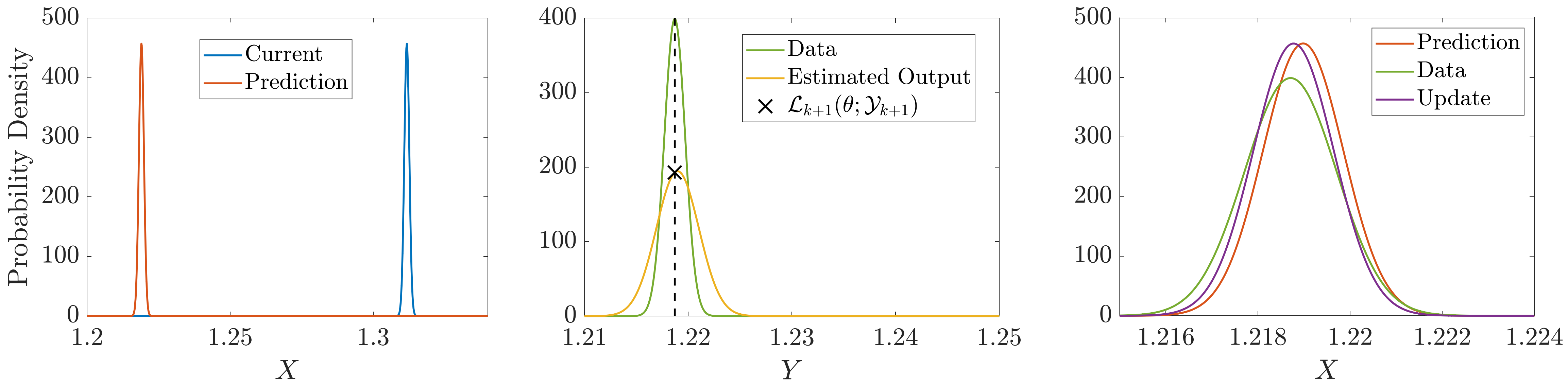
for $k = 0, \ldots, n-1$

Predict: $p(X_{k+1}|\mathcal{Y}_k, \theta) = \int p(X_{k+1}|X_k, \theta) p(X_k|\mathcal{Y}_k, \theta) dX_k$

Marginalize: $\mathcal{L}_{k+1}(\theta; \mathcal{Y}_{k+1}) = \int p(y_{k+1}|X_{k+1}, \theta) p(X_{k+1}|\mathcal{Y}_k, \theta) dX_{k+1}$

Update: $p(X_{k+1}|\mathcal{Y}_{k+1}, \theta) = \dfrac{p(y_{k+1}|X_{k+1}, \theta) p(X_{k+1}|\mathcal{Y}_k, \theta)}{p(y_{k+1}|\mathcal{Y}_k, \theta)}$

end for



Estimated outputs that fit the data and have low variance yield the largest marginal likelihood

Särkkä, S. (2013). *Bayesian filtering and smoothing* (No. 3). Cambridge University Press.

Marginal Likelihood
Regularization derived from first principles

Let the state be distributed normally as $X_k \sim \mathcal{N}(m_k, P_k)$

The negative log-likelihood is equivalent to a time-varying weighted least-squares objective with regularization

$$\mathcal{L}(\theta; \mathcal{Y}_n) \propto \sum_{k=1}^{n} \|y_k - H(\theta)m_k^-(\theta)\|^2_{S_k^{-1}(\theta)} + \log|2\pi S_k(\theta)|$$

Where

$$P_k^-(\theta) = A(\theta)P_{k-1}^+(\theta)A^T(\theta) + \Sigma(\theta)$$
$$S_k(\theta) = H(\theta)P_k^-(\theta)H^T(\theta) + \Gamma(\theta)$$

A dynamics matrix
H observation matrix

This objective prioritizes:

- low MSE: $\|y_k - H(\theta)m_k^-(\theta)\|^2_{S_k^{-1}(\theta)}$
- low sensitivity to state perturbations: $\log|2\pi S_k(\theta)|$

Outline

1. Existing approaches
2. Probabilistic formulation
3. Algorithm/Marginal likelihood
4. Results
5. Takeaways

ngalioto@umich.edu

Chaotic Duffing Oscillator: Formulation

$$\begin{bmatrix}\dot{x}\\ \ddot{x}\end{bmatrix} = \begin{bmatrix}0 & 1\\ \alpha & \delta\end{bmatrix}\begin{bmatrix}x\\ \dot{x}\end{bmatrix} + \beta\begin{bmatrix}0\\ x^3\end{bmatrix} + \begin{bmatrix}0\\ 1\end{bmatrix}\gamma\cos(\omega t),$$

$$y_k = x_k$$

We choose parameters that give a chaotic solution[1]

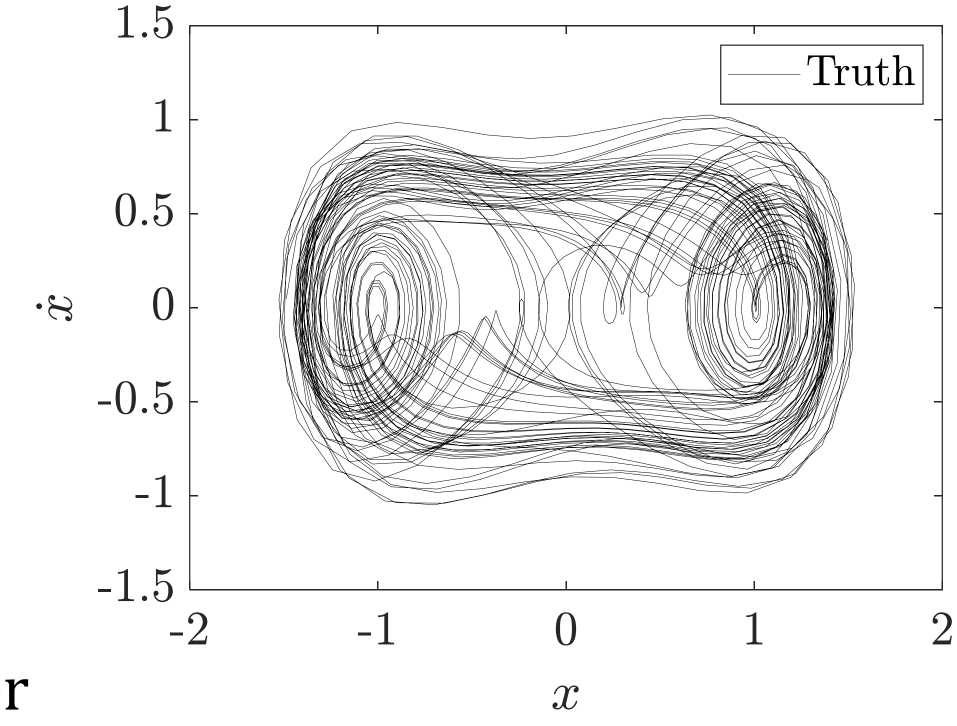
Model parametrization:

$$\mathbf{x}_0 = \mathbf{x}_0(\theta),\ d_{\mathbf{x}} = 2$$

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, u_k; \theta) + \xi_k, \qquad \xi_k \sim \mathcal{N}\big(0, \Sigma(\theta)\big)$$

$$y_k = [1 \quad 0]\mathbf{x}_k + \eta_k, \qquad \eta_k \sim \mathcal{N}(0, \Gamma)$$

The system possesses an underlying attractor



Neural network architecture[2]; 15 nodes/hidden layer

$$f(\mathbf{x}, u; \theta) = A_1(\theta)\tanh\left(A_2(\theta)\begin{bmatrix}\mathbf{x}\\ u\end{bmatrix} + b_2(\theta)\right) + A_3(\theta)\begin{bmatrix}\mathbf{x}\\ u\end{bmatrix} + b_3(\theta)$$

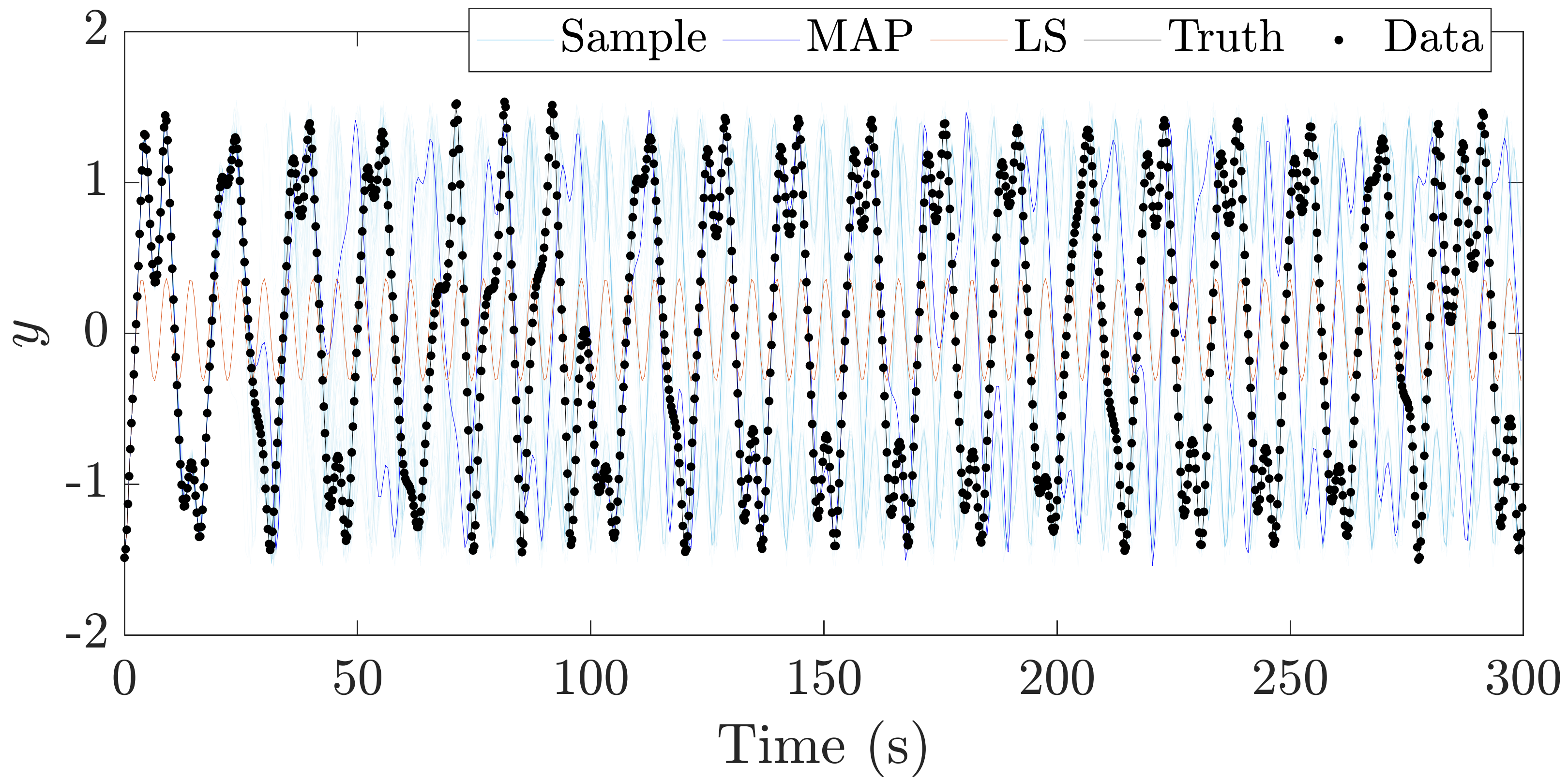
Weakly informative priors in order to emphasize the strength of the proposed likelihood

ngalioto@umich.edu

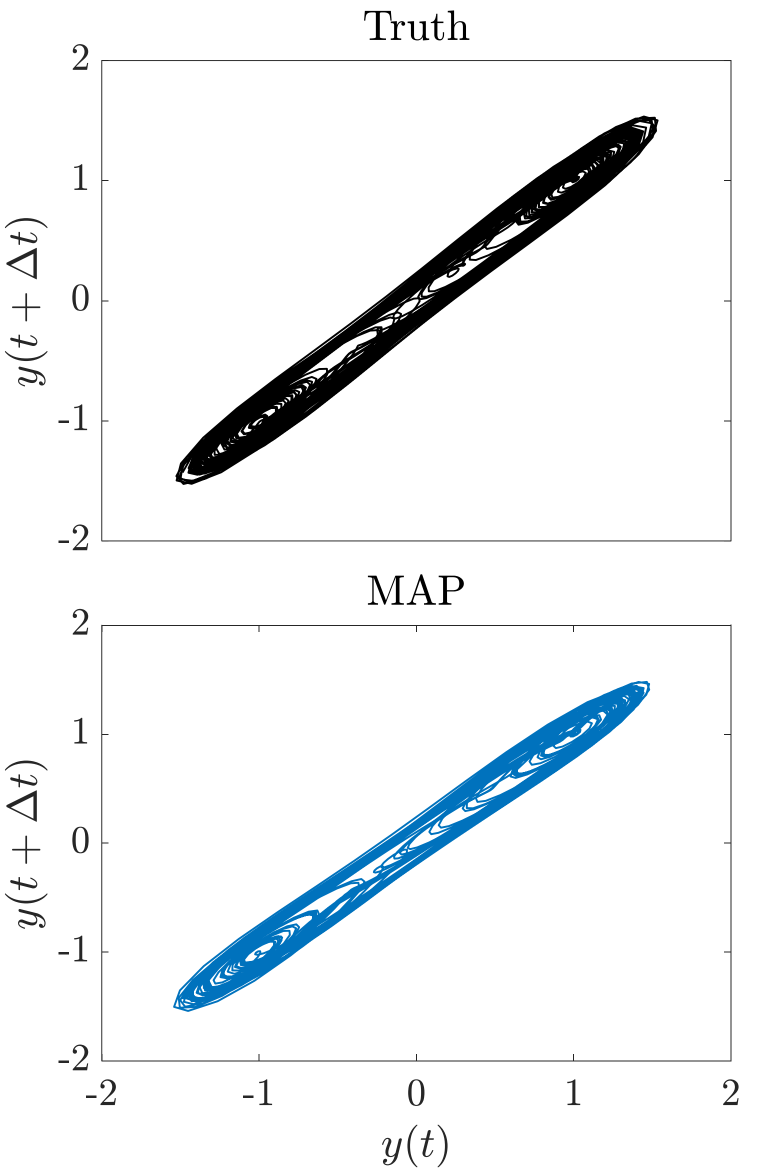
1. Jordan, D., & Smith, P. (2007). *Nonlinear ordinary differential equations: an introduction for scientists and engineers* (Vol. 10). Oxford University Press on Demand.
2. Beintema, G., Toth, R., & Schoukens, M. (2021, May). Nonlinear state-space identification using deep encoder networks. In *Learning for Dynamics and Control* (pp. 241-250). PMLR.

Chaotic Duffing Oscillator

The MAP estimate recovers the attractor despite having larger training MSE than the least squares (LS) estimate



$n = 1200$ data points over $T = 300$ seconds
Standard deviation of $\sigma = 10^{-3}$

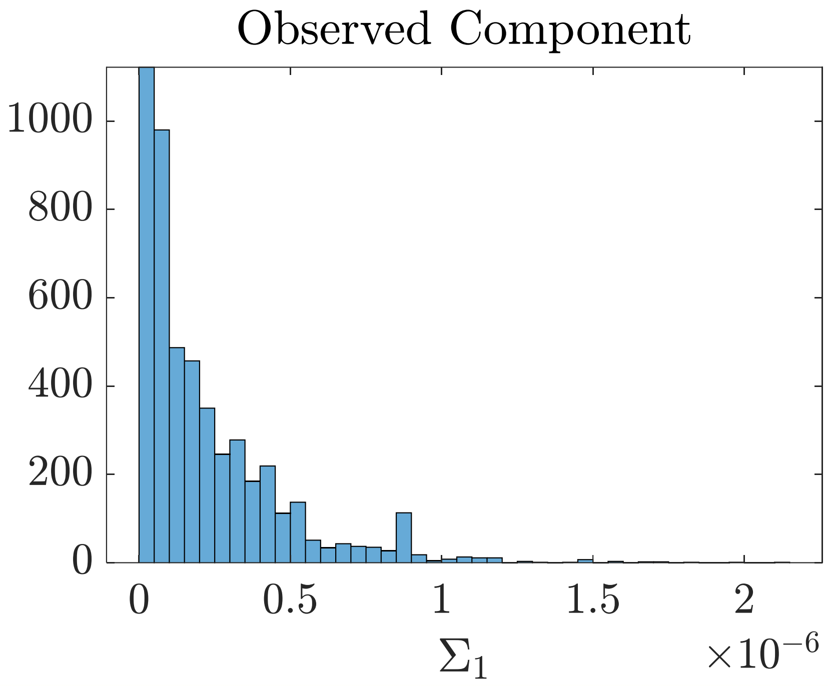


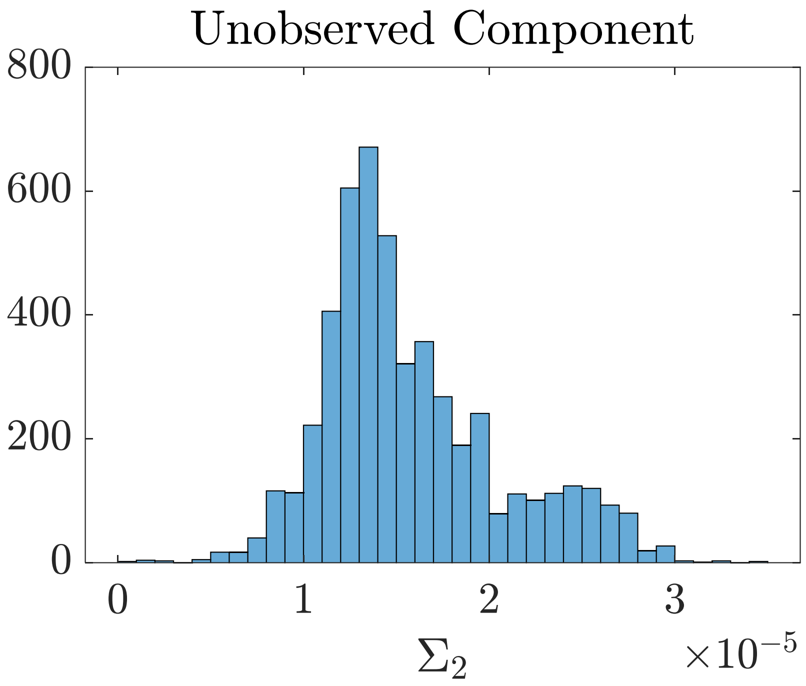
Phase space for twice the training time period

Chaotic Duffing Oscillator

Order of magnitude greater uncertainty in the dynamics of the unobserved variable

Marginals of process noise variance parameters:





PDE Quantity of Interest

Suppose we have a PDE system, but we are only interested in a low-dimensional quantity of interest (QoI)

Can we learn the dynamics of this QoI without modeling the full field?

ngalioto@umich.edu

Allen-Cahn Quantity of Interest (QoI)

1D PDE with forcing u, spatial coordinate $\xi \in [-1, 1]$ and time coordinate $t \in \mathbb{R}_+$

$$\frac{\partial w}{\partial t} = 0.2\frac{\partial^2 w}{\partial \xi^2} + w(1 - w^2) + \chi_{[-0.5,0.2]}(\xi)u(t)$$

χ is indicator function
Neumann boundary conditions

$$y_k = \int_{-1}^{1} w(\xi)^2 d\xi + \eta_k \qquad \eta_k \sim \mathcal{N}(0, 20^2)$$

101 measurements with $\Delta t = 0.10s$

Model parametrization:

$\mathbf{x}_0 = \mathbf{x}_0(\theta),\ d_{\mathbf{x}} = 8$

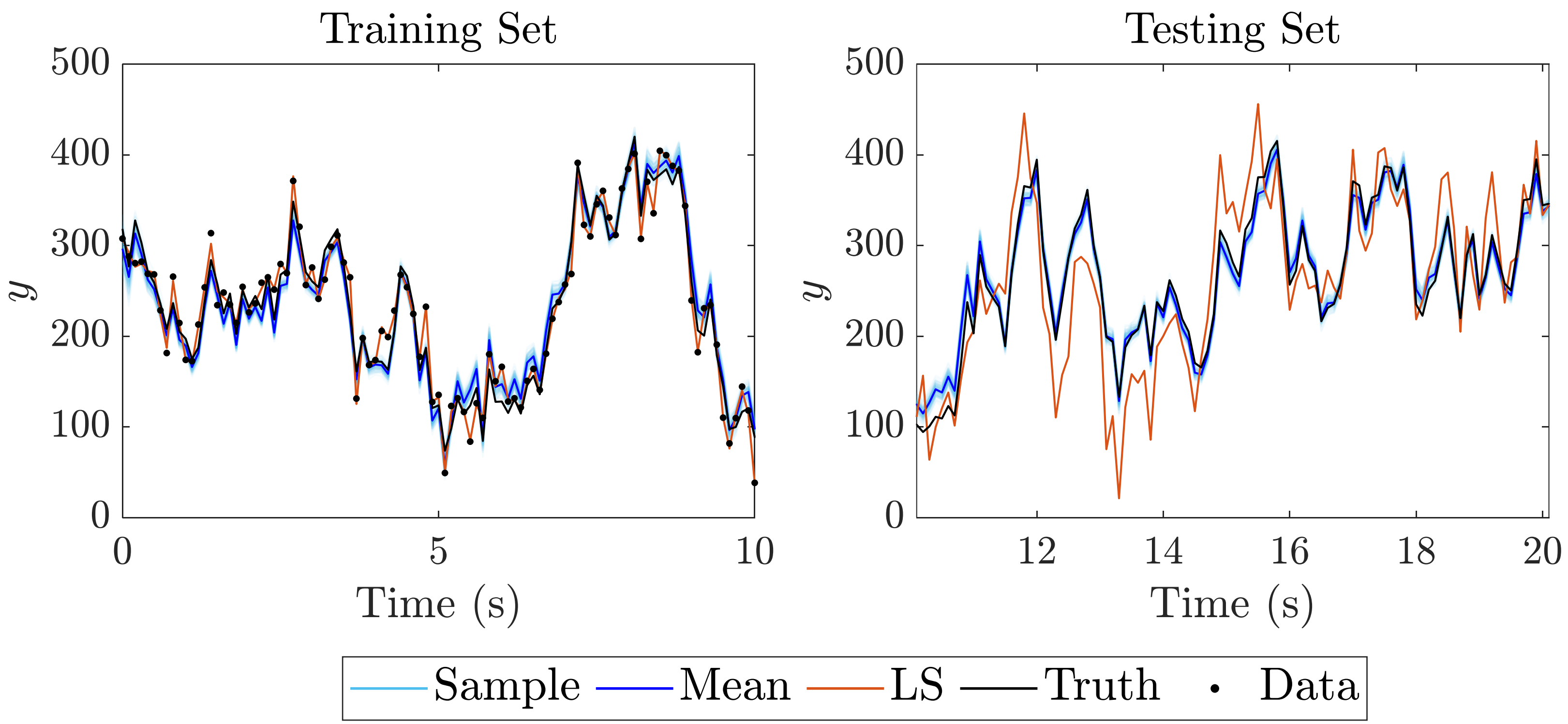
$\mathbf{x}_{k+1} = f(\mathbf{x}_k, u_k; \theta) + \xi_k, \qquad \xi_k \sim \mathcal{N}\big(0, \Sigma(\theta)\big)$

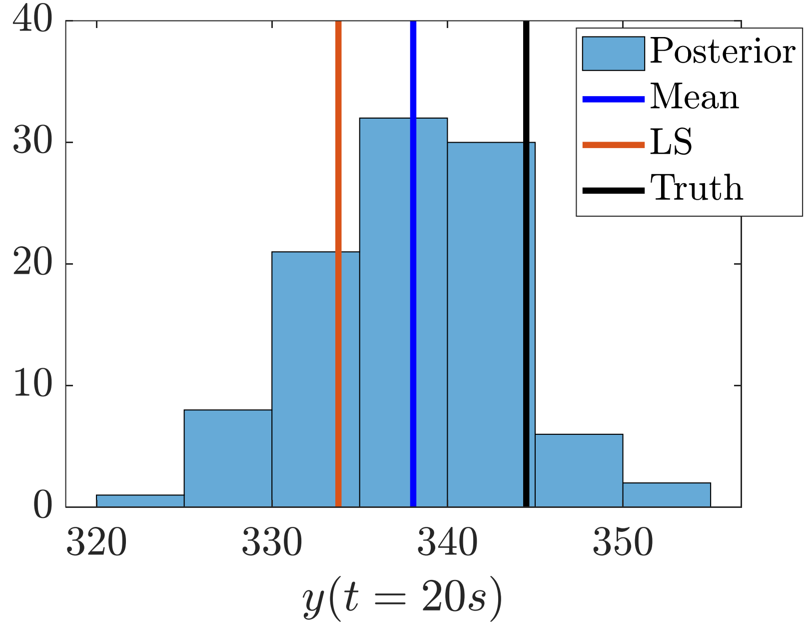
$y_k = [1 \quad \mathbf{0}_{1\times 7}]\mathbf{x}_k + \eta_k, \qquad \eta_k \sim \mathcal{N}\big(0, \Gamma(\theta)\big)$

Dolgov, S., Kalise, D., & Kunisch, K. K. (2021). Tensor Decomposition Methods for High-dimensional Hamilton--Jacobi--Bellman Equations. *SIAM Journal on Scientific Computing*, *43*(3), A1625-A1650.

Allen-Cahn Quantity of Interest (QoI)

The LS estimate overfits, but the inherent regularization of the Bayesian approach yields a more generalizable model





Testing MSE

Posterior Mean	72.35
Least Squares	3,404.

Outline

Main Takeaways

- Optimally accounting for different types of uncertainty can lead to robustness even for chaotic systems
- Modeling deterministic systems with stochastic models introduces built-in regularization and optimization benefits

Related Works

1. Galioto, N., & Gorodetsky, A. A. (2020). Bayesian system ID: optimal management of parameter, model, and measurement uncertainty. *Nonlinear Dynamics*, *102*(1), 241-267.
2. Galioto, N., & Gorodetsky, A. A. (2021). A New Objective for Identification of Partially Observed Linear Time-Invariant Dynamical Systems from Input-Output Data. In *Learning for Dynamics and Control* (pp. 1180-1191). PMLR.

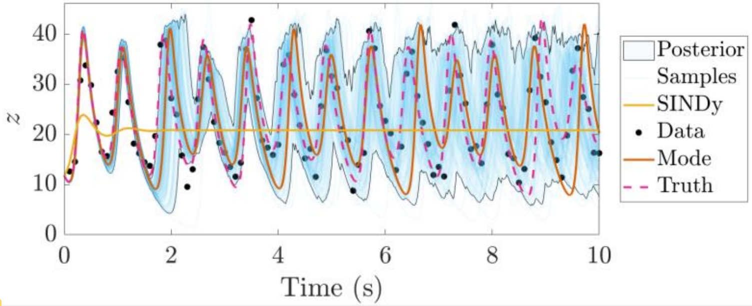
Funding

- DARPA Physics of AI Program
  - "Physics Inspired Learning and Learning the Order and Structure of Physics."
- AFOSR Program in Computational Mathematics

Appendix

Results: Lorenz '63

Accounting for model error enhances robustness

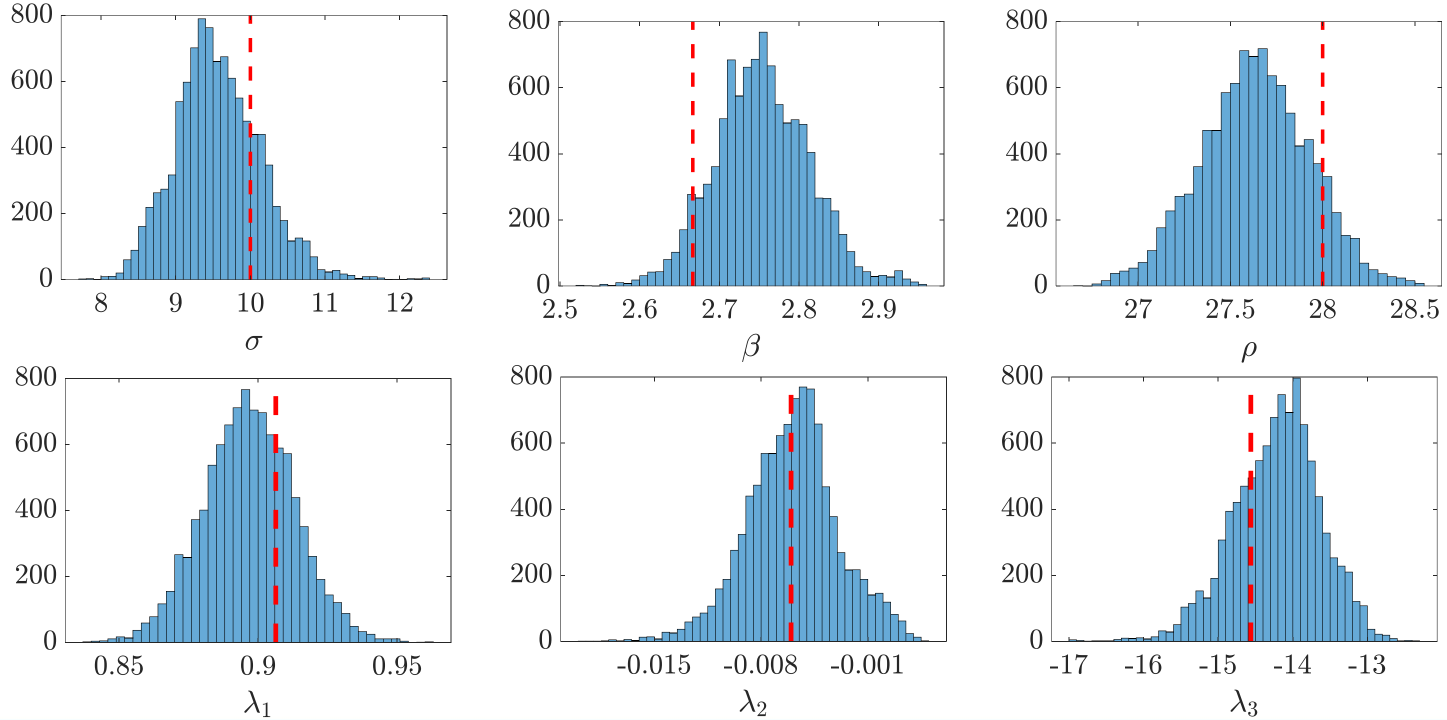


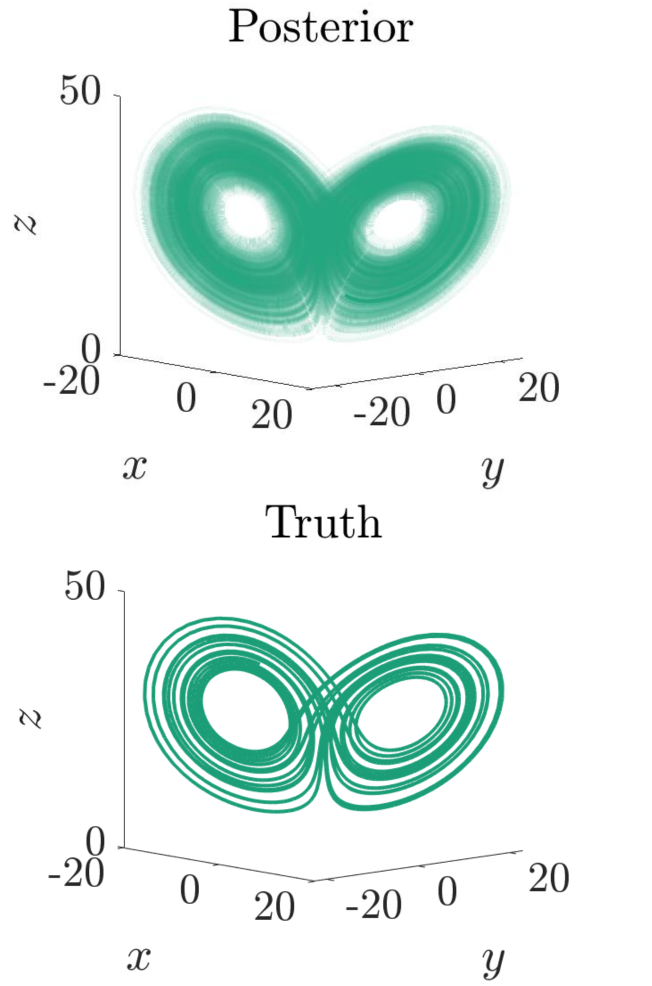
Most positive Lyapunov exponent: $\lambda_1 = 0.906$

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

Recent works[1,2,3] commonly use:
$n = 300$
$\Delta t = 0.01s$
$\sigma_R = 0.0$

$n = 100$
$\Delta t = 0.10s$
$\sigma_R = 2.0$





ngalioto@umich.edu

1. Lazzús, J. A., Rivera, M., & López-Caraballo, C. H. (2016). Parameter estimation of Lorenz chaotic system using a hybrid swarm intelligence algorithm. *Physics Letters A*, *380*(11-12), 1164-1171.
2. Xu, S., Wang, Y., & Liu, X. (2018). Parameter estimation for chaotic systems via a hybrid flower pollination algorithm. *Neural Computing and Applications*, *30*(8), 2607-2623.
3. Zhuang, L., Cao, L., Wu, Y., Zhong, Y., Zhangzhong, L., Zheng, W., & Wang, L. (2020). Parameter Estimation of Lorenz Chaotic System Based on a Hybrid Jaya-Powell Algorithm. *IEEE Access*, *8*, 20514-20522.

Hamiltonian Systems

In mechanical systems, the Hamiltonian $\mathcal{H}$ is the sum of potential energy U and kinetic energy T

$$\mathcal{H}(q,p) = T(q,p) + U(q,p)$$

q generalized position
p generalized momentum

Equations of motion are derived from the Hamiltonian

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

Hamiltonian systems have a number of physical properties

- Conservation
- Reversibility
- Symplecticness

Dynamical Model Parameterization

Ensures the learned system is Hamiltonian

$$\mathcal{H}(q, p, \theta_\Psi) = \frac{1}{2} p^T p + U(q, \theta_\Psi)$$

Differentiation

$$\dot{q} = p, \qquad \dot{p} = -\frac{\partial U(q, \theta_\Psi)}{\partial q}$$

Leapfrog Method

Conserves Hamiltonian and preserves symplectic structure throughout evaluation

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

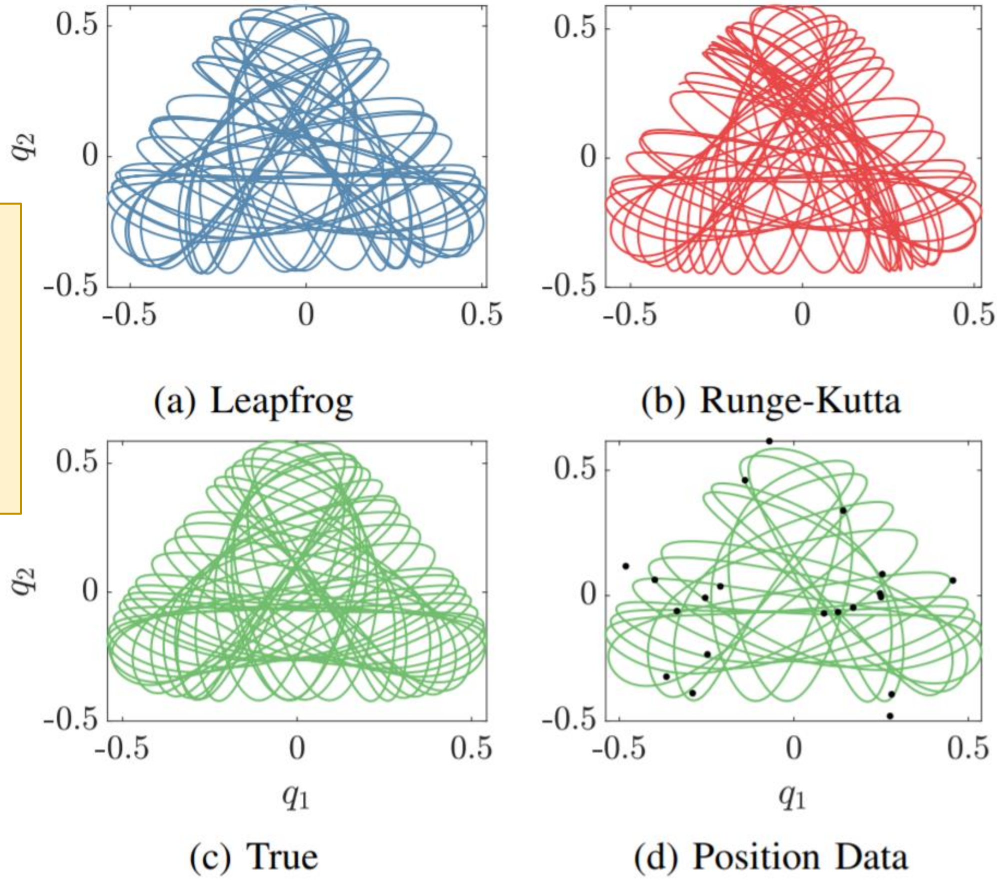
Results: Hénon-Heiles

The symplectic approach learns a more accurate Hamiltonian

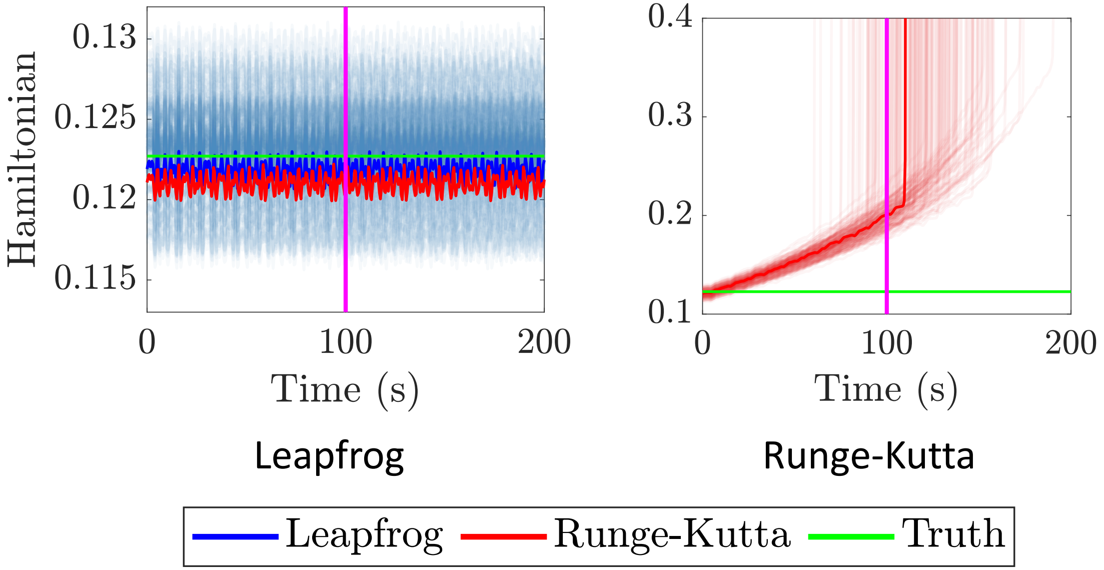
Truth: $U(q_1, q_2) = \frac{1}{2}q_1^2 + \frac{1}{2}q_2^2 + q_1^2 q_2 - \frac{1}{3}q_2^3$

Data Generation:
- $n = 20$
- $\Delta t = 5$
- $\sigma = 0.05$

Phase plots



Hamiltonian over time



Leapfrog Runge-Kutta

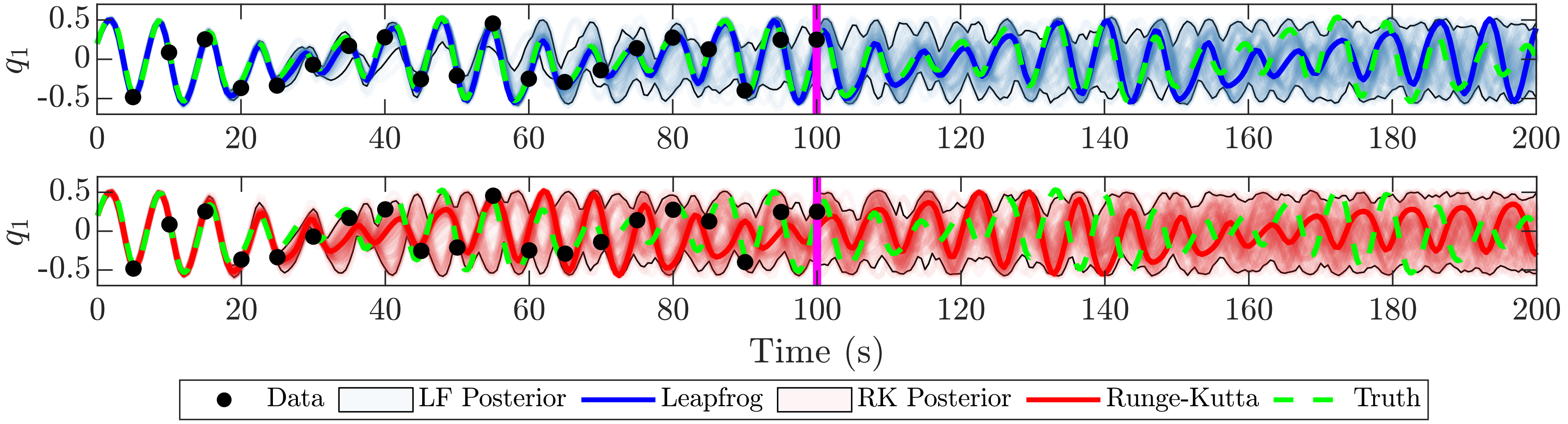
The method equipped with RK must learn a smaller Hamiltonian to compensate for being non-conservative

Relative mean error:
Leapfrog: 0.7%; Runge-Kutta: 1.3%

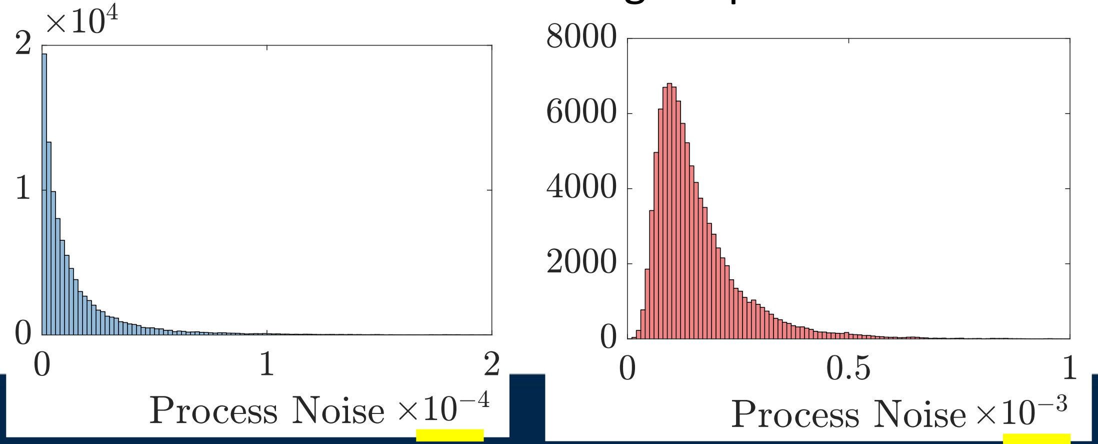
Results: Hénon-Heiles

The symplectic approach yields greater certainty

Posterior estimates of q_1 trajectory



Process noise marginal posteriors



Leapfrog

Runge-Kutta

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