Learning dynamics on invariant measures using PDE-constrained optimization

SIAM NNP — Contributed Talk

Jonah Botvinick-Greenhouse
Center for Applied Mathematics
Cornell University
Ithaca, NY 14850

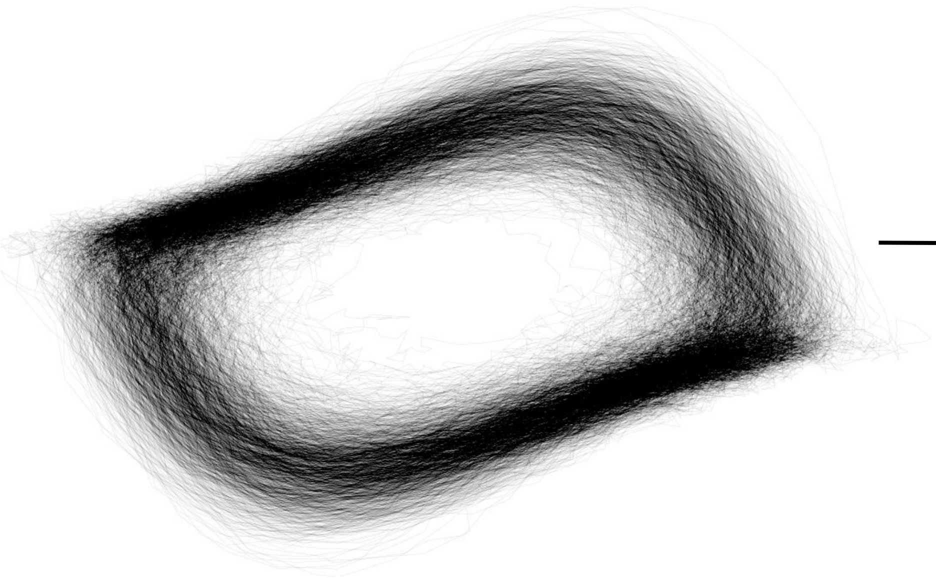
Yunan Yang
Department of Mathematics
Cornell University
Ithaca, NY 14850

Acknowledgment: Robert Martin, U.S. Army Research Office

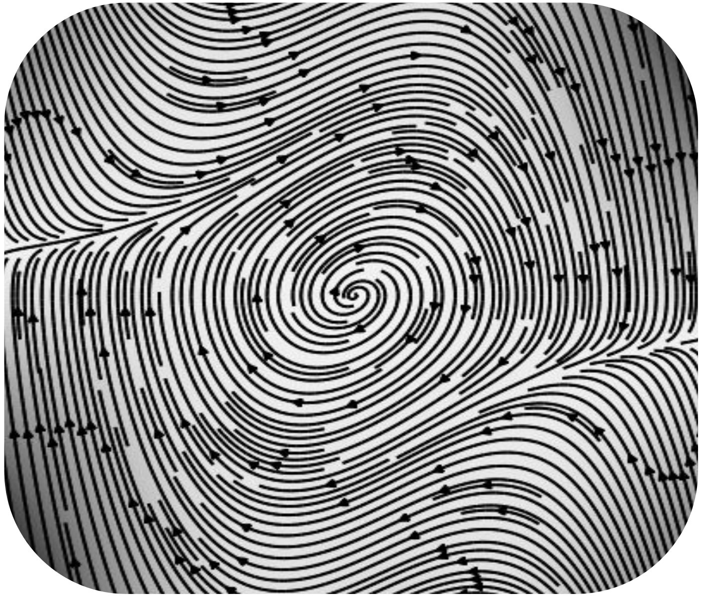
Background & motivation

What does it mean to learn dynamics?

$$\{\widetilde{x}(t_i)\}$$



$$dX_t = v(X_t)dt + \sqrt{2D}dW_t$$



Imposing strict assumptions on the data quality

Noisy measurements, non-uniform in time, sampled slowly

$$\{\widetilde{x}(t_i)\}$$

$\longrightarrow$

Equations of motion with stochastic forcing

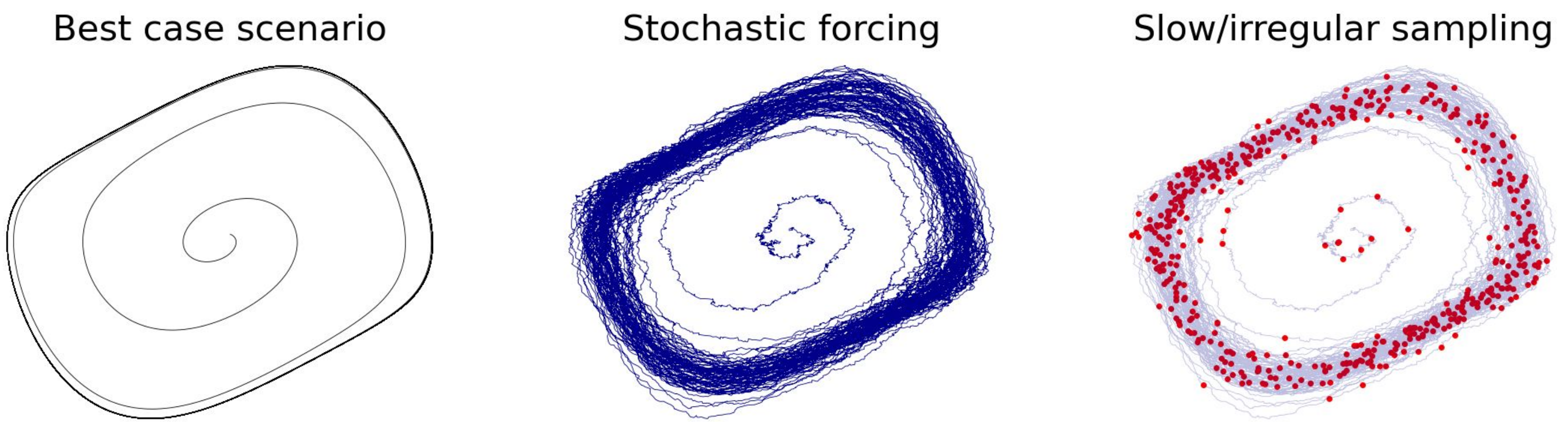
$$dX_t = v(X_t)dt + \sqrt{2D}dW_t$$

Imposing strict assumptions on the data quality

Noisy measurements, non-uniform in time, sampled slowly

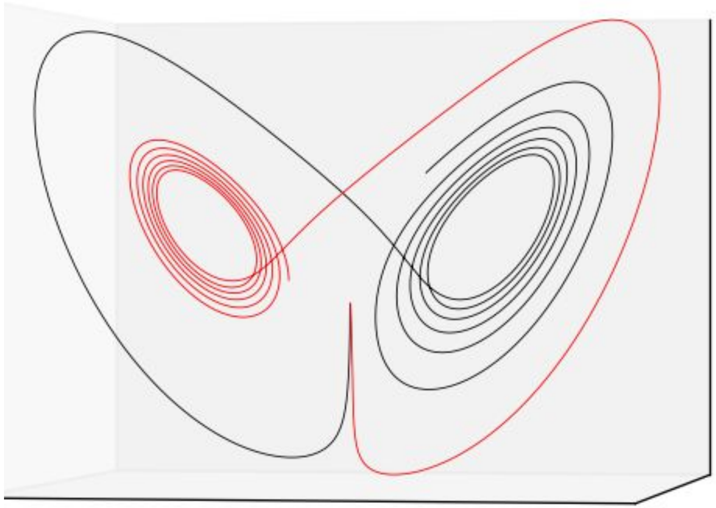
$$\{\widetilde{x}(t_i)\} \longrightarrow dX_t = v(X_t)dt + \sqrt{2D}dW_t$$

Equations of motion with stochastic forcing



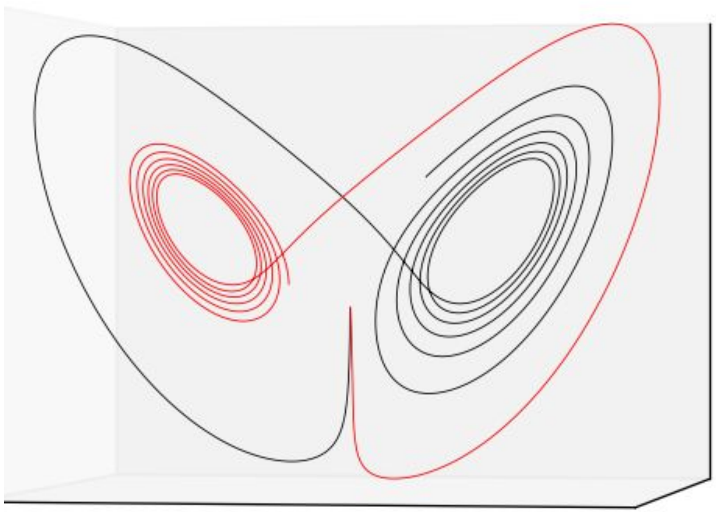
Lagrangian vs. Eulerian dynamics

Lagrangian - describes trajectories of individual particles

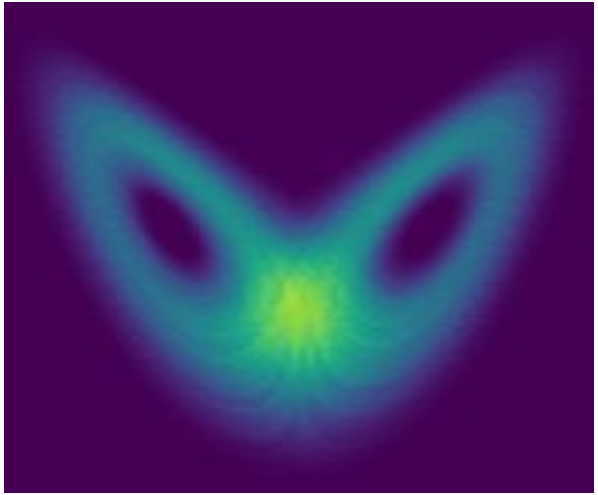


Lagrangian vs. Eulerian dynamics

Lagrangian - describes trajectories of individual particles



Eulerian - describes the distribution of all particles



How do we go from Lagrangian to Eulerian?

Study the statistical properties of trajectories.

$$\mu_{x,N}(B) = \frac{1}{N}\sum_{k=0}^{N-1} \chi_B(T^k(x))$$

How do we go from Lagrangian to Eulerian?

Study the statistical properties of trajectories.

$$\mu_{x,N}(B) = \frac{1}{N}\sum_{k=0}^{N-1} \chi_B(T^k(x))$$

When does this provide information about “many” trajectories?

$$m(\{x \in \Omega : \mu_{x,N} \to^* \mu\}) > 0$$

How do we go from Lagrangian to Eulerian?

Study the statistical properties of trajectories.

$$\mu_{x,N}(B) = \frac{1}{N}\sum_{k=0}^{N-1} \chi_B(T^k(x))$$

When does this provide information about "many" trajectories?

$$m(\{x \in \Omega : \mu_{x,N} \to^* \mu\}) > 0$$

A weak-* limit of occupation measures is invariant.

$$\mu(T^{-1}(B)) = \mu(B)$$

What is the goal?

Invert the mapping $v \mapsto \mu$

What is the goal?

Invert the mapping $v \mapsto \mu$

Previous work:

- Greve, C., Hara, K., Martin, R., Eckhardt, D. & Koo, J. A data-driven approach to model calibration for nonlinear dynamical systems. *Journal Of Applied Physics*. **125** (2019)
- Yang, Y., Nurbekyan, L., Negrini, E., Martin, R. & Pasha, M. Optimal transport for parameter identification of chaotic dynamics via invariant measures. *SIAM Journal On Applied Dynamical Systems*. **22**, 269-310 (2023)

What is the goal?

Invert the mapping $v \mapsto \mu$

Previous work:

- Greve, C., Hara, K., Martin, R., Eckhardt, D. & Koo, J. A data-driven approach to model calibration for nonlinear dynamical systems. *Journal Of Applied Physics*. **125** (2019)
- Yang, Y., Nurbekyan, L., Negrini, E., Martin, R. & Pasha, M. Optimal transport for parameter identification of chaotic dynamics via invariant measures. *SIAM Journal On Applied Dynamical Systems*. **22**, 269-310 (2023)

New contributions:

- ❑ Ability to model intrinsic noise
- ❑ Large-scale parameter identification
- ❑ Learning dynamics in delay coordinates
- ❑ Uncertainty quantification

Building a forward model

The PDE forward model

$$\underbrace{\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho v) = D\nabla^2 \rho}_{\text{Eulerian}} \iff \underbrace{dX_t = v(X_t)dt + \sqrt{2D}dW_t}_{\text{Lagrangian}}$$

Physical measures describe the long term statistical behavior of Lagrangian trajectories, so we use stationary solutions of the Fokker-Planck Equation (FPE) as a surrogate model.

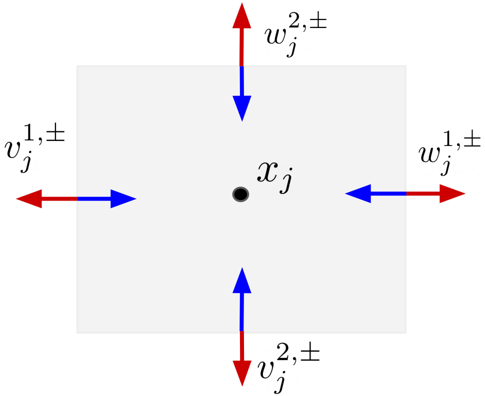
The PDE forward model

$$\underbrace{\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho v) = D\nabla^2 \rho}_{\text{Eulerian}} \iff \underbrace{dX_t = v(X_t)dt + \sqrt{2D}dW_t}_{\text{Lagrangian}}$$

Physical measures describe the long term statistical behavior of Lagrangian trajectories, so we use stationary solutions of the Fokker-Planck Equation (FPE) as a surrogate model.

We discretize the FPE via a first order upwind finite volume method to form a Markov chain approximation of the dynamics.

$$\rho^{(\ell+1)} = M\rho^{(\ell)}, \quad M = I + K$$



A closer look at the discretization...

$$K_i := \begin{bmatrix}
\ddots & & & & \\
 & -v^{i,-}_{j-1} + \frac{D}{\Delta x_i} & & & \\
\ddots & \vdots & & & \\
 & v^{i,-}_{j-1} - w^{i,+}_{j-1} - \frac{2D}{\Delta x_i} & -v^{i,-}_{j} + \frac{D}{\Delta x_i} & & \\
\ddots & \vdots & \vdots & & \\
 & w^{i,+}_{j-1} + \frac{D}{\Delta x_i} & v^{i,-}_{j} - w^{i,+}_{j} - \frac{2D}{\Delta x_i} & -v^{i,-}_{j+1} + \frac{D}{\Delta x_i} & \\
 & & \vdots & \vdots & \ddots \\
 & & w^{i,+}_{j} + \frac{D}{\Delta x_i} & v^{i,-}_{j+1} - w^{i,+}_{j+1} - 2\frac{D}{\Delta x_i} & \\
 & & & \vdots & \ddots \\
 & & & w^{i,+}_{j+1} + \frac{D}{\Delta x_i} & \\
 & & & & \ddots
\end{bmatrix} \in \mathbb{R}^{N \times N}.$$

Finding the Markov chain’s steady state

We will use the Markov chain’s steady state as a model for the underlying physical measure

$$M\rho = \rho$$

Finding the Markov chain's steady state

We will use the Markov chain's steady state as a model for the underlying physical measure

$$M\rho = \rho$$

Some potential difficulties:

- ❏ If the diffusion is zero, the stationary distribution may not be unique.
- ❏ The transition matrix may be ill-conditioned.
- ❏ We may want to learn the velocity away from the attractor.

Finding the Markov chain's steady state

We will use the Markov chain's steady state as a model for the underlying physical measure

$$M\rho = \rho$$

Some potential difficulties:

- ❑ If the diffusion is zero, the stationary distribution may not be unique.
- ❑ The transition matrix may be ill-conditioned.
- ❑ We may want to learn the velocity away from the attractor.

Solution: Teleportation regularization from Google's Pagerank algorithm.

$$M_\epsilon := (1-\epsilon)M + \frac{\epsilon}{N}\mathbf{1}\mathbf{1}^\top \in \mathbb{R}^{N\times N}, \qquad \mathbf{1} := \begin{bmatrix}1 & \dots & 1\end{bmatrix} \in \mathbb{R}^N.$$

Finding the Markov chain's steady state

We will use the Markov chain's steady state as a model for the underlying physical measure

$$M\rho = \rho$$

Some potential difficulties:

- ❏ If the diffusion is zero, the stationary distribution may not be unique.
- ❏ The transition matrix may be ill-conditioned.
- ❏ We may want to learn the velocity away from the attractor.

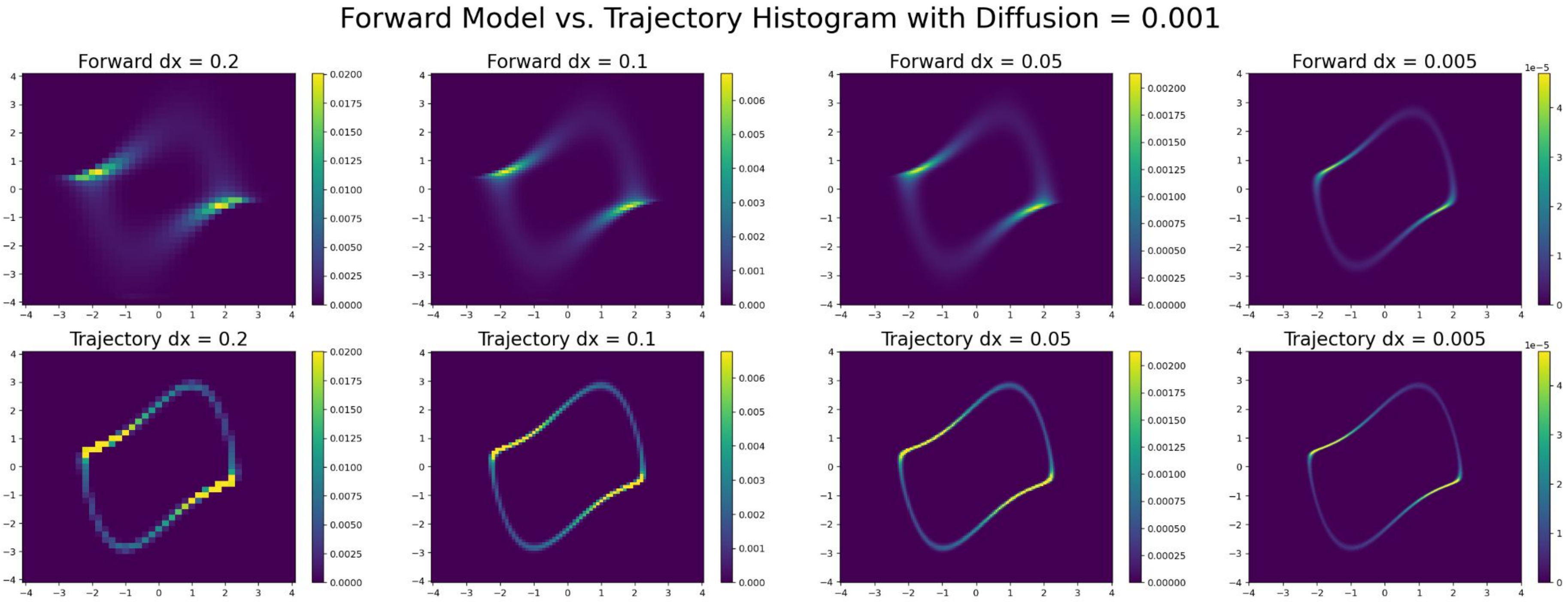
Solution: Teleportation regularization from Google's Pagerank algorithm.

$$M_\epsilon := (1-\epsilon)M + \frac{\epsilon}{N}\mathbf{1}\mathbf{1}^\top \in \mathbb{R}^{N\times N}, \qquad \mathbf{1} := \begin{bmatrix}1 & \dots & 1\end{bmatrix} \in \mathbb{R}^N.$$

Now, the steady state can be uniquely found by solving a sparse linear system.

$$(1-\epsilon)(M-I)\rho = -\frac{\epsilon}{N}\mathbf{1}$$

Forward model vs. occupation measures



Choice of objective function

$$L^2(\rho, \rho^*) := \frac{1}{2}\int_\Omega (\rho(x) - \rho^*(x))^2 dx$$

Squared L2 Norm

$$D_{\mathrm{KL}}(\rho, \rho^*) := \int_{\Omega'} \rho^*(x) \log\left(\frac{\rho^*(x)}{\rho(x)}\right) dx$$

Kullback-Leibler Divergence

$$D_{\mathrm{JS}}(\rho, \rho^*) = \frac{1}{2} D_{KL}(\rho, \rho') + \frac{1}{2} D_{KL}(\rho^*, \rho')$$

Jenson-Shannon Divergence

$$W_2^2(\rho, \rho^*) := \inf_{T_{\rho,\rho^*} \in \mathcal{M}} \int_\Omega |x - T_{\rho,\rho^*}(x)|^2 d\rho(x)$$

Quadratic Wasserstein Distance

Computing the gradient

The adjoint state method

Compute the Fréchet derivative of the objective function with respect to the current density

$$\boxed{\frac{\partial L_2}{\partial \rho} = \rho - \rho^*.}$$

$$\boxed{\frac{\partial D_{\mathrm{KL}}}{\partial \rho} = -\frac{\rho^*(x)}{\rho(x)}.}$$

$$\boxed{\frac{\partial D_{\mathrm{JS}}}{\partial \rho} = \frac{1}{2}\log\left(\frac{2\rho}{\rho+\rho^*}\right)}$$

$$\boxed{\frac{\partial W_2^2}{\partial \rho} = \phi,}$$

The adjoint state method

Compute the Fréchet derivative of the objective function with respect to the current density

$$\boxed{\frac{\partial L_2}{\partial \rho} = \rho - \rho^*.}$$

$$\boxed{\frac{\partial D_{\mathrm{KL}}}{\partial \rho} = -\frac{\rho^*(x)}{\rho(x)}.}$$

$$\boxed{\frac{\partial D_{\mathrm{JS}}}{\partial \rho} = \frac{1}{2}\log\left(\frac{2\rho}{\rho + \rho^*}\right)}$$

$$\boxed{\frac{\partial W_2^2}{\partial \rho} = \phi,}$$

Solve the adjoint equation

$$(M_\epsilon - I)^T \lambda = -\left(\frac{\partial \mathcal{J}}{\partial \rho} - \frac{\partial \mathcal{J}}{\partial \rho} \cdot \rho \mathbf{1}\right)^T$$

The adjoint state method

Compute the Fréchet derivative of the objective function with respect to the current density

$$\boxed{\frac{\partial L_2}{\partial \rho} = \rho - \rho^*.}$$

$$\boxed{\frac{\partial D_{\mathrm{KL}}}{\partial \rho} = -\frac{\rho^*(x)}{\rho(x)}.}$$

$$\boxed{\frac{\partial D_{\mathrm{JS}}}{\partial \rho} = \frac{1}{2}\log\left(\frac{2\rho}{\rho+\rho^*}\right)}$$

$$\boxed{\frac{\partial W_2^2}{\partial \rho} = \phi,}$$

Solve the adjoint equation

$$(M_\epsilon - I)^T \lambda = -\left(\frac{\partial \mathcal{J}}{\partial \rho} - \frac{\partial \mathcal{J}}{\partial \rho} \cdot \rho \mathbf{1}\right)^T$$

Compute gradient with respect to the piecewise constant velocities used in the Markov matrix.

$$\frac{\partial \mathcal{J}}{\partial v_i} = \lambda \cdot \frac{\partial M_\epsilon}{\partial v_i} \rho$$

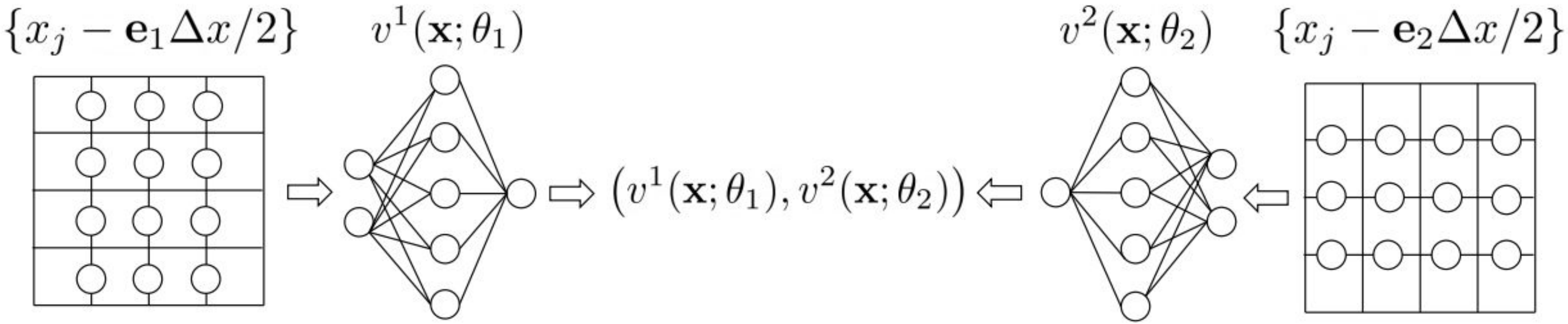
Velocity parameterization

$$v = v(\theta) \implies \frac{\partial \mathcal{J}}{\partial \theta_k} = \frac{\partial \mathcal{J}}{\partial v} \cdot \frac{\partial v}{\partial \theta_k}$$

Velocity parameterization

$$v = v(\theta) \implies \frac{\partial \mathcal{J}}{\partial \theta_k} = \frac{\partial \mathcal{J}}{\partial v} \cdot \frac{\partial v}{\partial \theta_k}$$

We considered three parameterizations: piecewise constant, global polynomial, and neural network.



The optimization framework

1.) Solve the forward problem $$M_\epsilon \rho = \rho$$

2.) Evaluate the cost $$\mathcal{J}(\rho, \rho^*)$$

3.) Compute the Frechet derivative $$\phi = \frac{\partial \mathcal{J}}{\partial \rho}$$

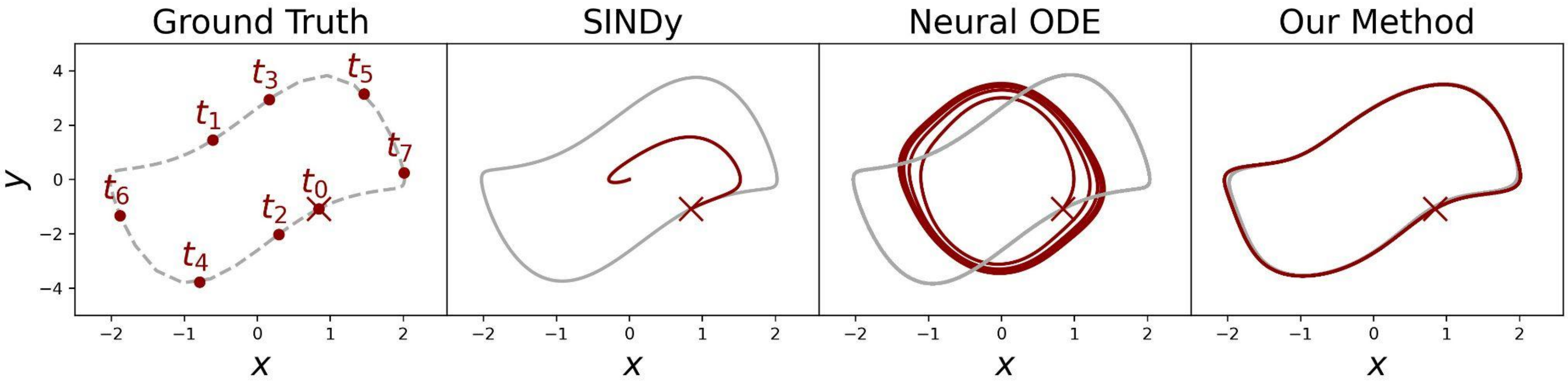
4.) Solve the adjoint equation $$(M_\epsilon^T - I)\lambda = -\phi + \phi \cdot \rho \mathbf{1}$$

5.) Compute the gradient $$\frac{\partial \mathcal{J}}{\partial v_i} = \lambda \cdot \frac{\partial M_\epsilon}{\partial v_i} \rho \qquad \frac{\partial \mathcal{J}}{\partial \theta_k} = \frac{\partial \mathcal{J}}{\partial v} \cdot \frac{\partial v}{\partial \theta_k}$$

6.) Descend $$\text{Adam, L-BFGS-B, CG, etc.}$$

Numerical Results

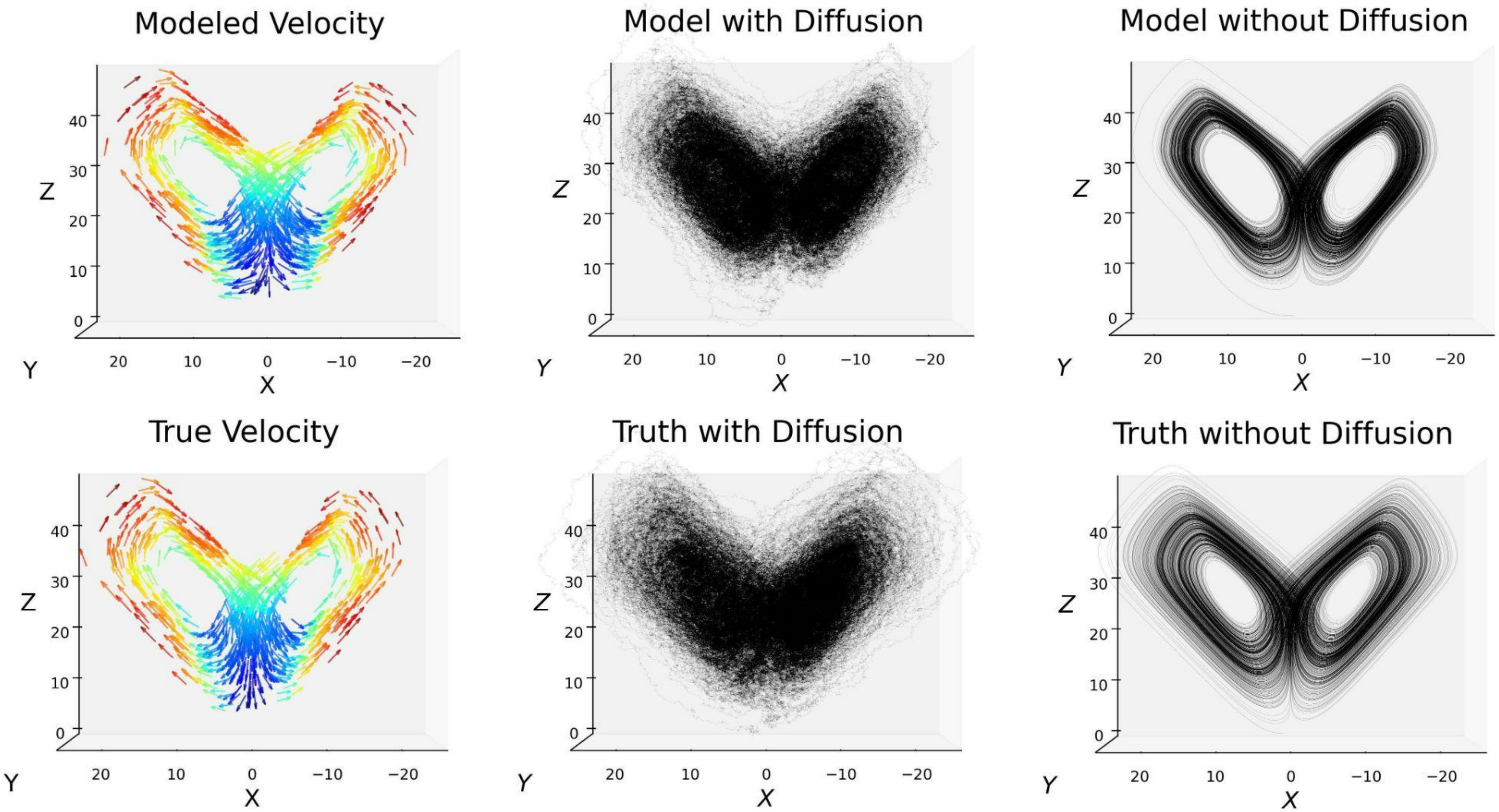
Comparison with other methods



Method	Sampling Freq.	Wall-Clock Time (s)	Error
SINDy	10.00	$2 \cdot 10^{-2}$	$5.6 \cdot 10^{-3}$
Neural ODE	10.00	$5 \cdot 10^{2}$	$5.32 \cdot 10^{-3}$
Ours	10.00	$5 \cdot 10^{2}$	$1.14 \cdot 10^{-1}$

Method	Sampling Freq.	Wall-Clock Time (s)	Error
SINDy	0.25	10^{-2}	3.52
Neural ODE	0.25	$5 \cdot 10^{2}$	1.81
Ours	0.25	$5 \cdot 10^{2}$	$6.79 \cdot 10^{-2}$

Application to a chaotic system



What if we only have partial observations?

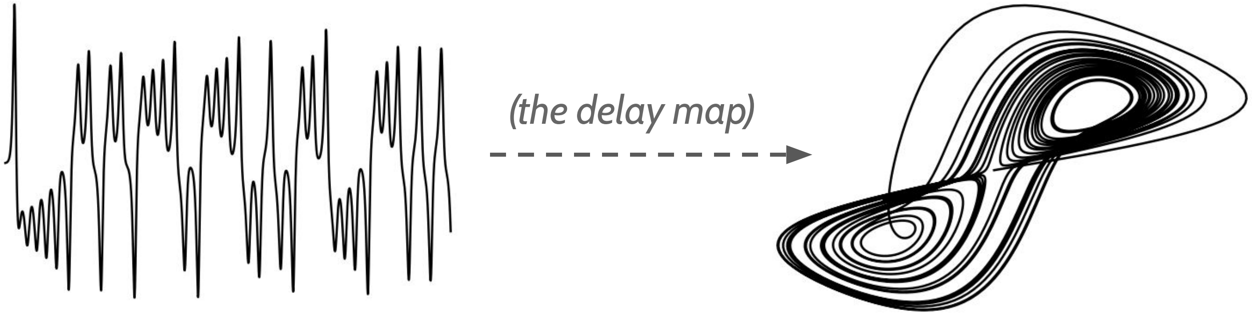
$$\underbrace{\dot{x} = v(x)}_{\text{full dynamics}} \qquad \underbrace{y : \mathcal{M} \to \mathbb{R}}_{\text{observation function}} \qquad \underbrace{\{y(x(t_i))\}_{i=1}^{N}}_{\text{available data}}$$

What if we only have partial observations?

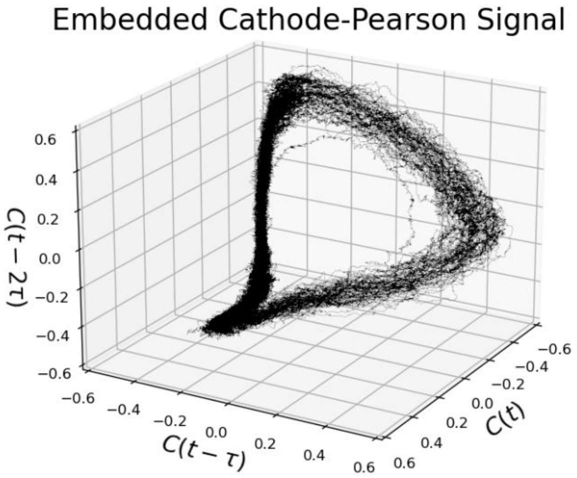
$$\underbrace{\dot{x} = v(x)}_{\text{full dynamics}} \qquad \underbrace{y : \mathcal{M} \to \mathbb{R}}_{\text{observation function}} \qquad \underbrace{\{y(x(t_i))\}_{i=1}^{N}}_{\text{available data}}$$

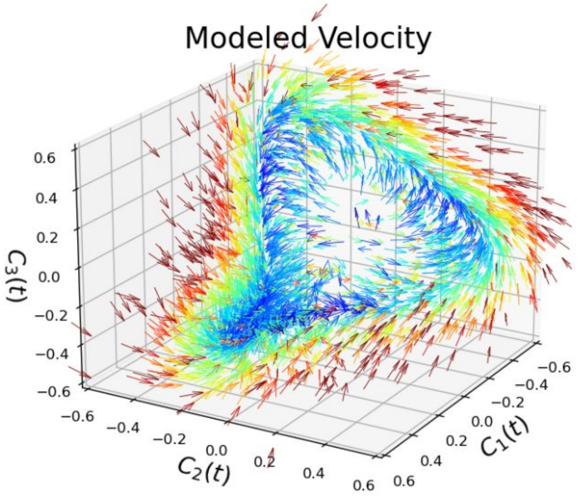
Motivated by Takens' Theorem (1981), we can instead learn the dynamics in delay coordinates.

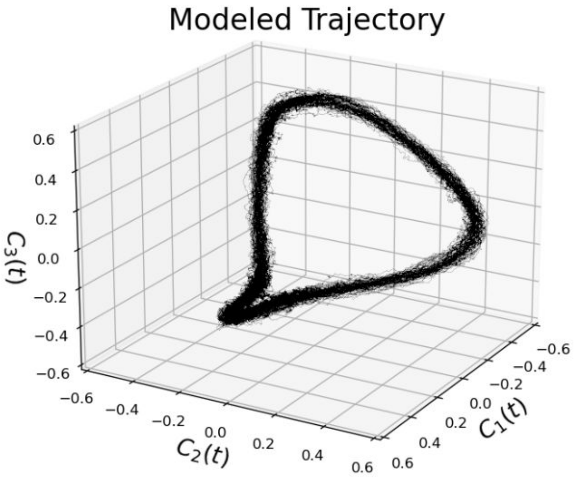
$$\exists\, \underbrace{\Phi : \mathcal{M} \to \mathbb{R}^d}_{\text{diffeomorphism}} \text{ with } \Phi(x(t)) = (y(t), y(t-\tau), \ldots, y(t-2d\tau))$$



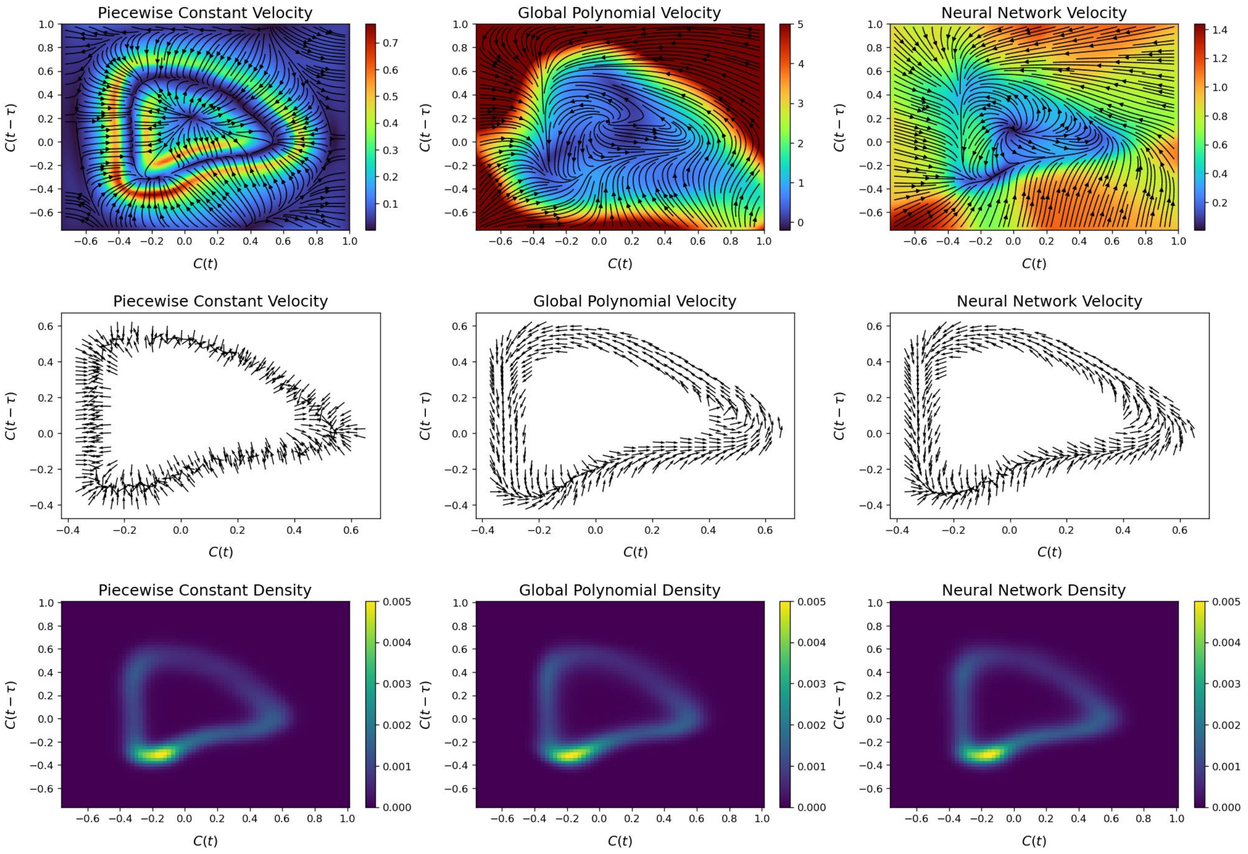
Application to a Hall-effect thruster



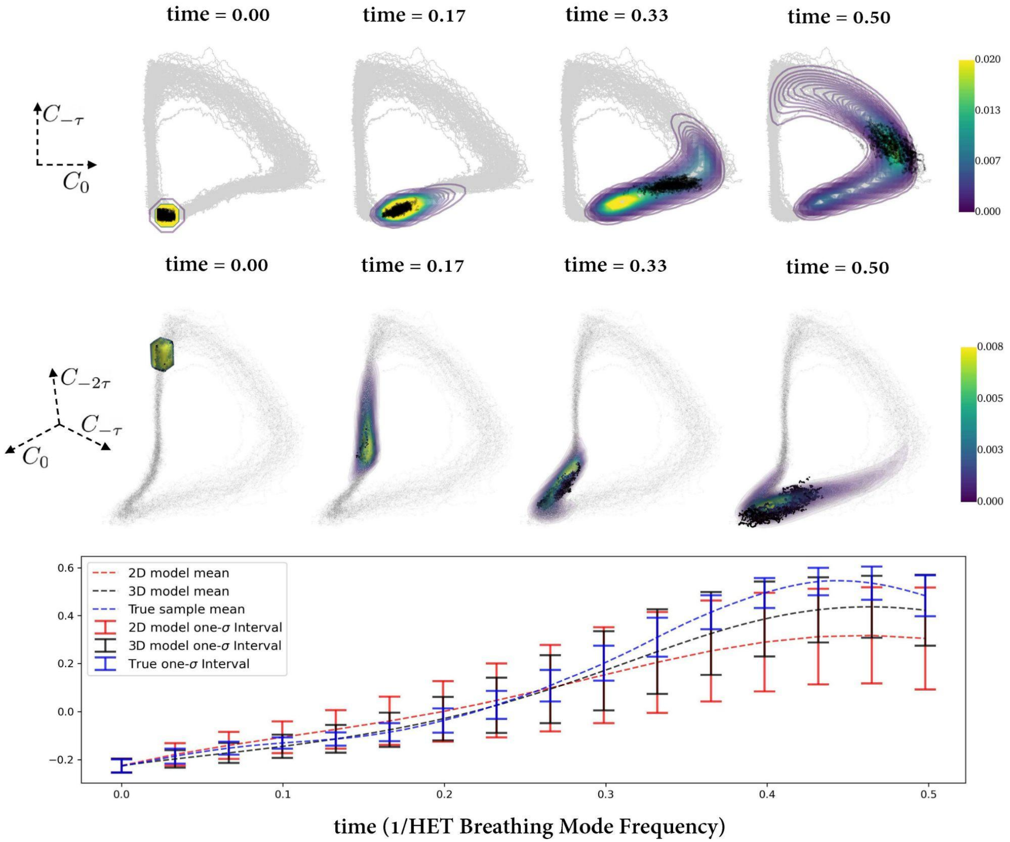




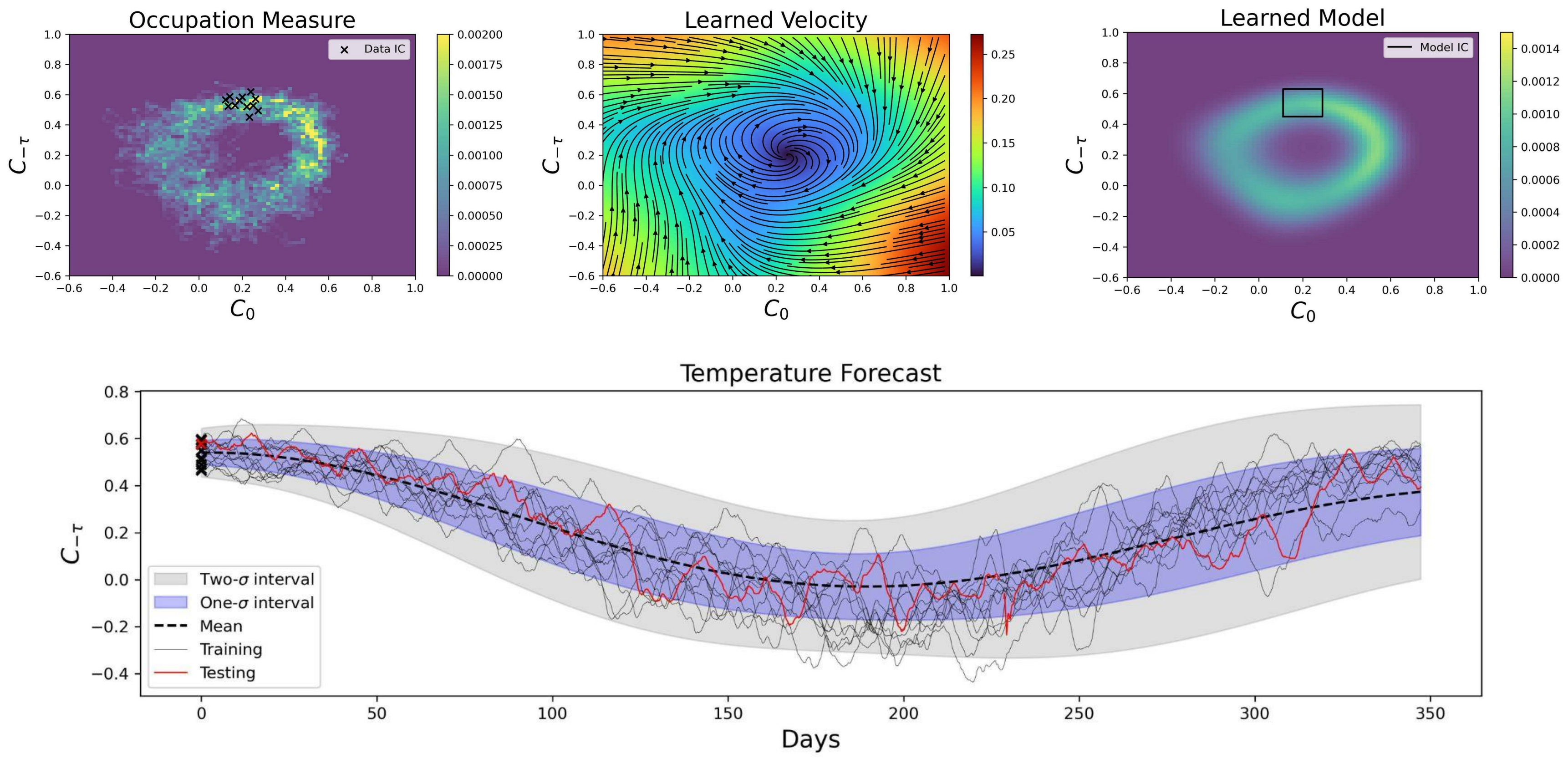
Varying the parameterization



Quantifying model uncertainty



Temperature prediction with uncertainty quantification



Conclusion

- We reformulated ODE/SDE learning as a PDE-constrained optimization.

Conclusion

- ❏ We reformulated ODE/SDE learning as a PDE-constrained optimization.
- ❏ We used the adjoint state method to obtain a fast gradient calculation.

Conclusion

- ❑ We reformulated ODE/SDE learning as a PDE-constrained optimization.
- ❑ We used the adjoint state method to obtain a fast gradient calculation.
- ❑ We then used gradient-based methods to perform model identification.

Conclusion

- We reformulated ODE/SDE learning as a PDE-constrained optimization.
- We used the adjoint state method to obtain a fast gradient calculation.
- We then used gradient-based methods to perform model identification.
- The learned models were used to perform uncertainty quantification.

Conclusion

- ❑ We reformulated ODE/SDE learning as a PDE-constrained optimization.
- ❑ We used the adjoint state method to obtain a fast gradient calculation.
- ❑ We then used gradient-based methods to perform model identification.
- ❑ The learned models were used to perform uncertainty quantification.

Thank you!