Learning Dynamical Systems from Invariant Measures

BIRS Workshop: New Ideas in Computational Inverse Problems

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

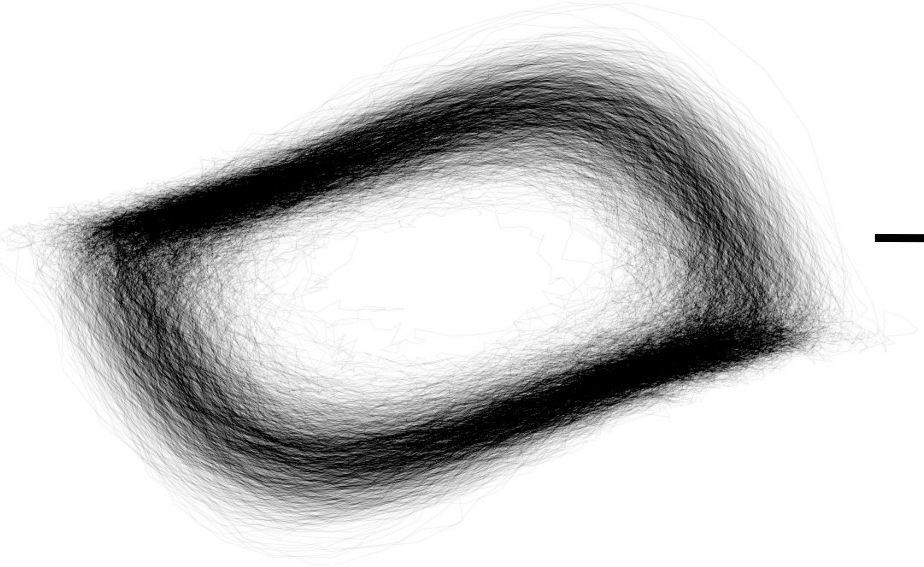
Yunan Yang
Institute for Theoretical Studies
ETH Zurich
Zurich, Switzerland 8092

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

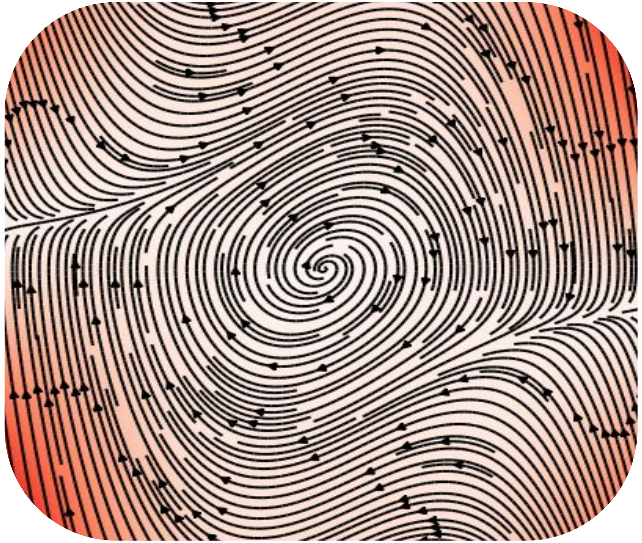
Motivation and Theory

What does it mean to learn a system?

Data



Computer Model



Imposing strict assumptions on the data quality

Noisy measurements, non-uniform in time, sampled slowly

Equations of motion with stochastic forcing

$$\{\widetilde{x}(t_i)\} \longrightarrow \dot{x} = v(x) + \omega(x,t)$$

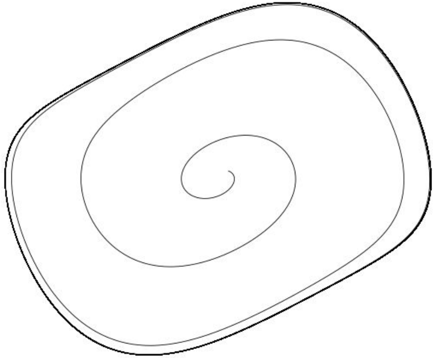
Imposing strict assumptions on the data quality

Noisy measurements, non-uniform in time, sampled slowly

Equations of motion with stochastic forcing

$$\{\widetilde{x}(t_i)\} \longrightarrow \dot{x} = v(x) + \omega(x,t)$$

Best case scenario



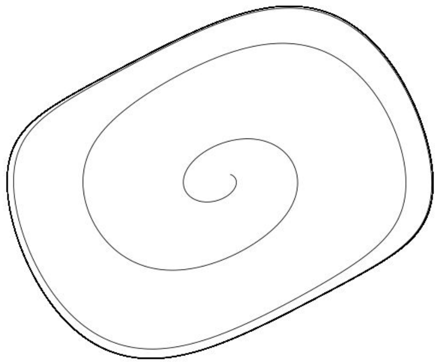
Imposing strict assumptions on the data quality

Noisy measurements, non-uniform in time, sampled slowly

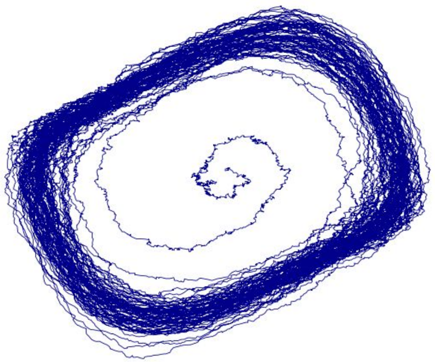
Equations of motion with stochastic forcing

$$\{\widetilde{x}(t_i)\} \longrightarrow \dot{x} = v(x) + \omega(x,t)$$

Best case scenario



Stochastic forcing

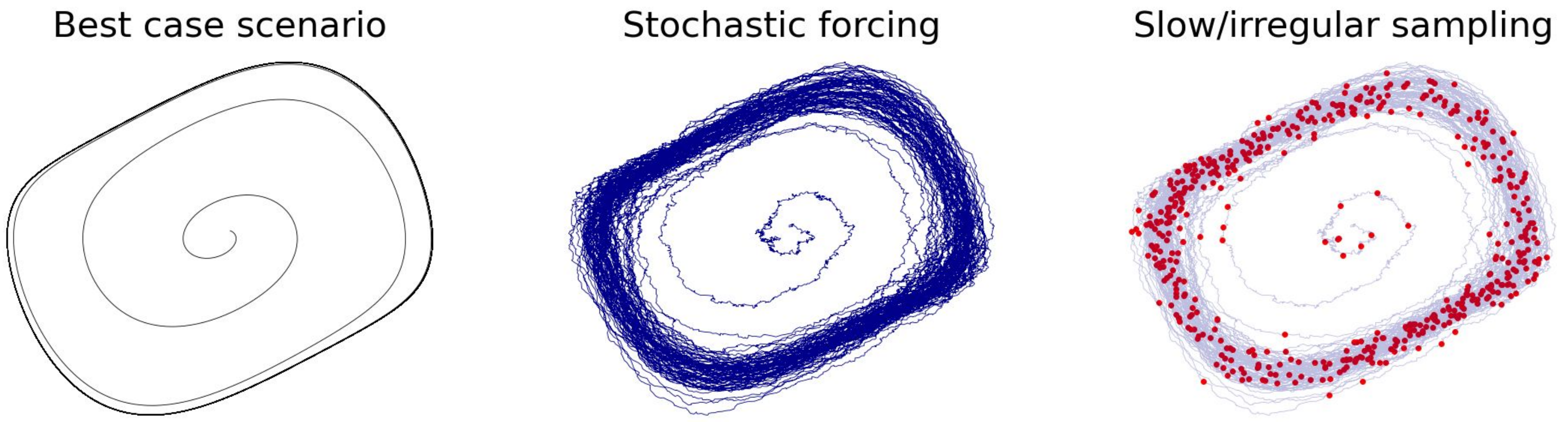


Imposing strict assumptions on the data quality

Noisy measurements, non-uniform in time, sampled slowly

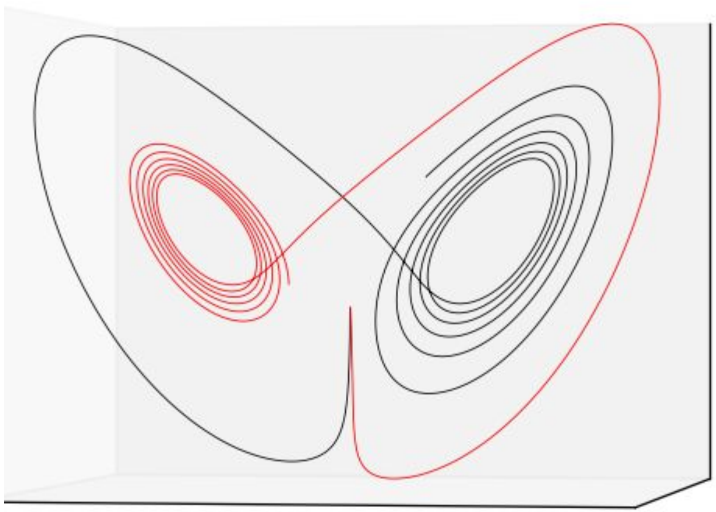
Equations of motion with stochastic forcing

$$\{\widetilde{x}(t_i)\} \longrightarrow \dot{x} = v(x) + \omega(x, t)$$



Lagrangian vs. Eulerian Dynamics

Lagrangian - describes trajectories of individual particles

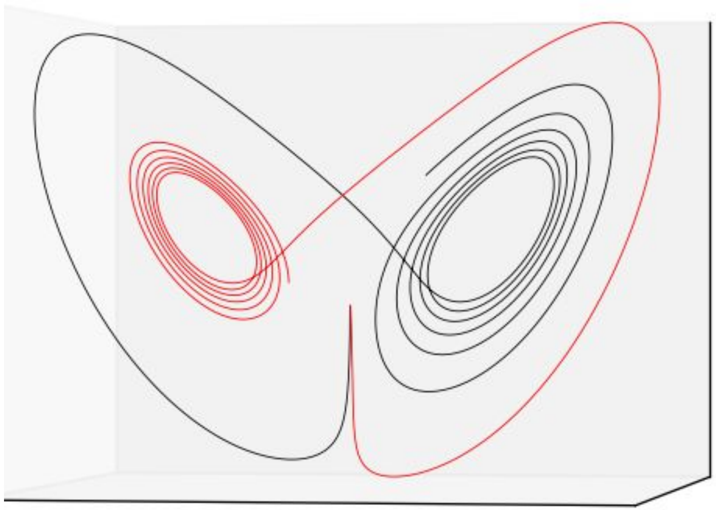


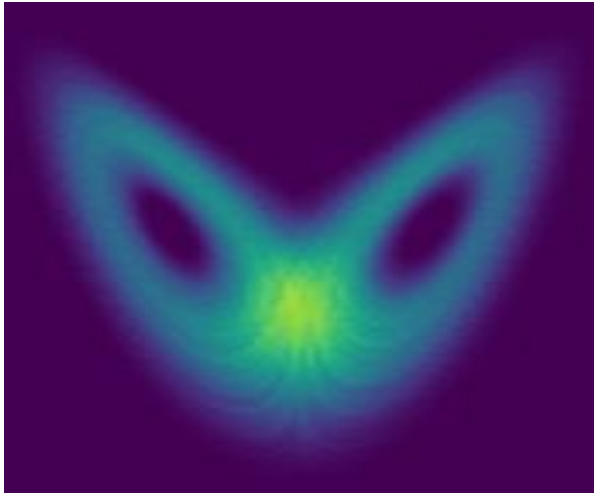
Eulerian - describes the distribution of all particles

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!

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

Occupation measures characterizes the the average time spent in a measurable set B for the initial condition x.

How do we go from Lagrangian to Eulerian?

Study the statistical properties of trajectories!

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

Occupation measures characterizes the the average time spent in a measurable set B for the initial condition x.

When does this provide information about "many" trajectories?

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

When such a measure μ exists, it is said to be physical.

How do we go from Lagrangian to Eulerian?

Study the statistical properties of trajectories!

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

Occupation measures characterizes the the average time spent in a measurable set B for the initial condition x.

When does this provide information about “many” trajectories?

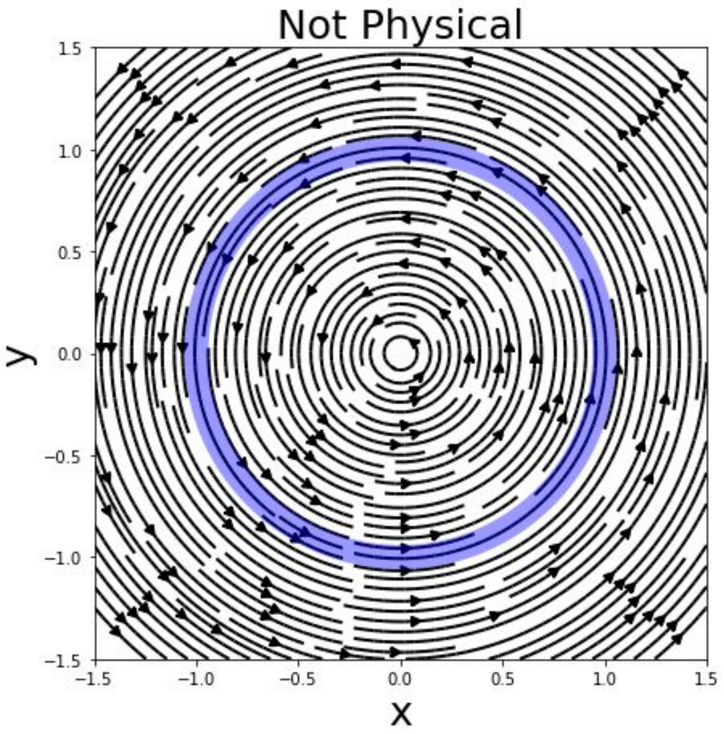
$$m(\{x \in \Omega : \mu_{x,N} \to^* \mu\}) > 0$$

When such a measure μ exists, it is said to be physical.

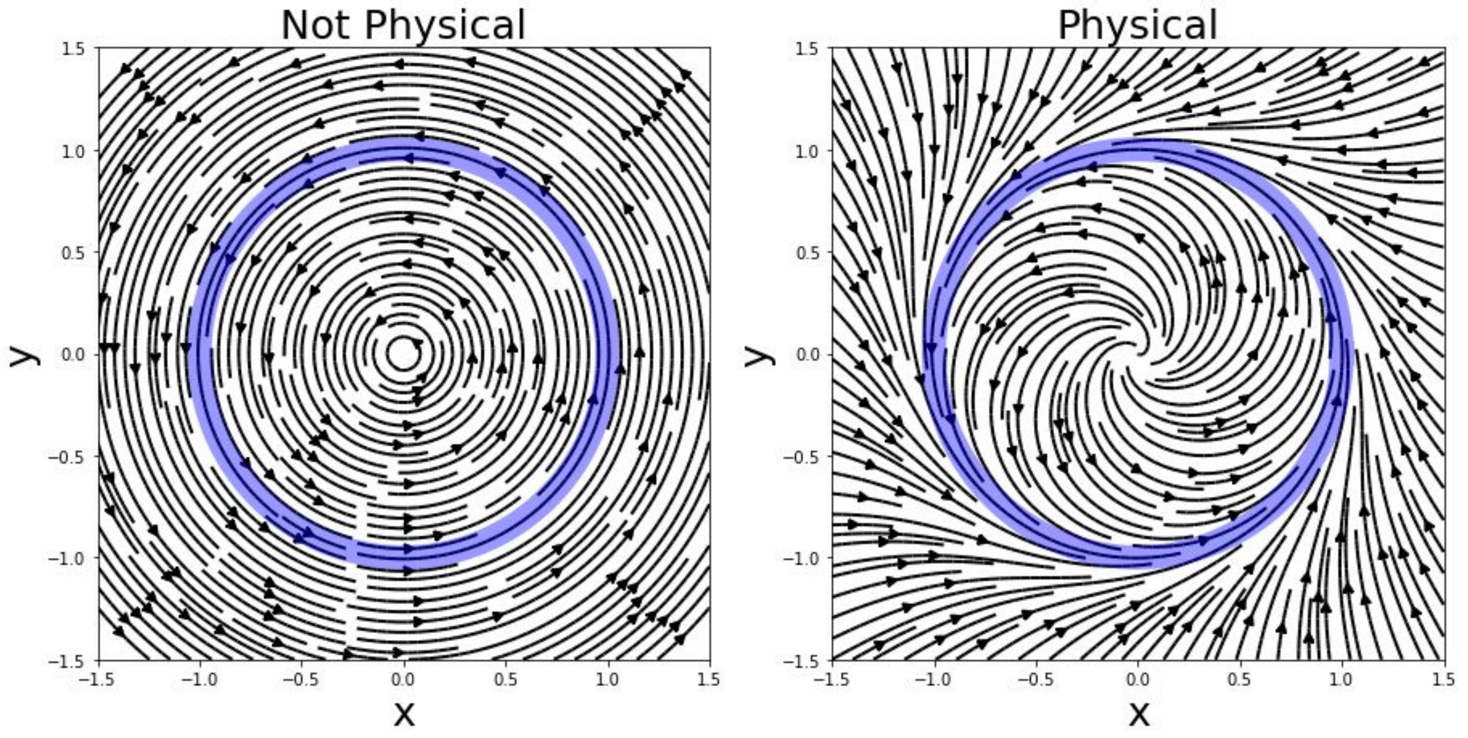
A weak-* limit of occupation measures is invariant.

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

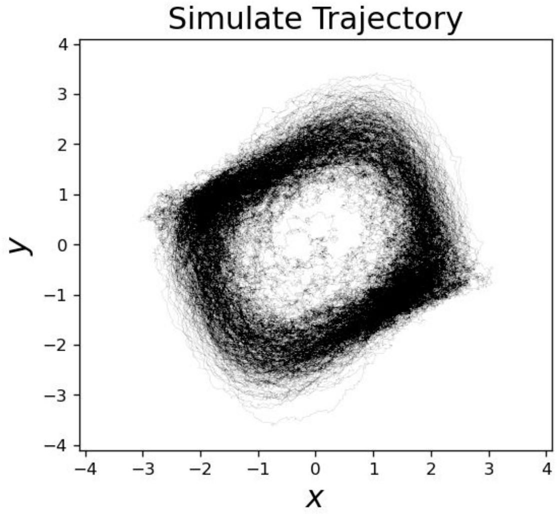
Example and Non-Example of a Physical Measure



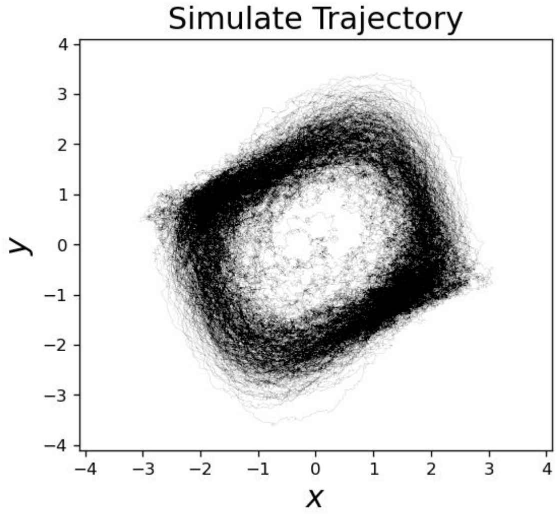
Example and Non-Example of a Physical Measure

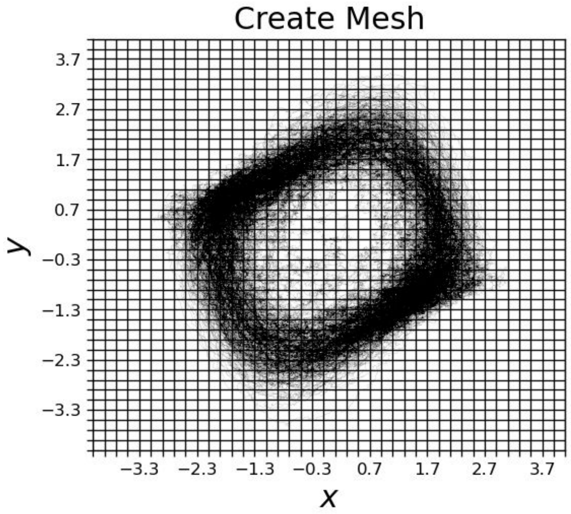


How do we computationally approximate a physical measure?

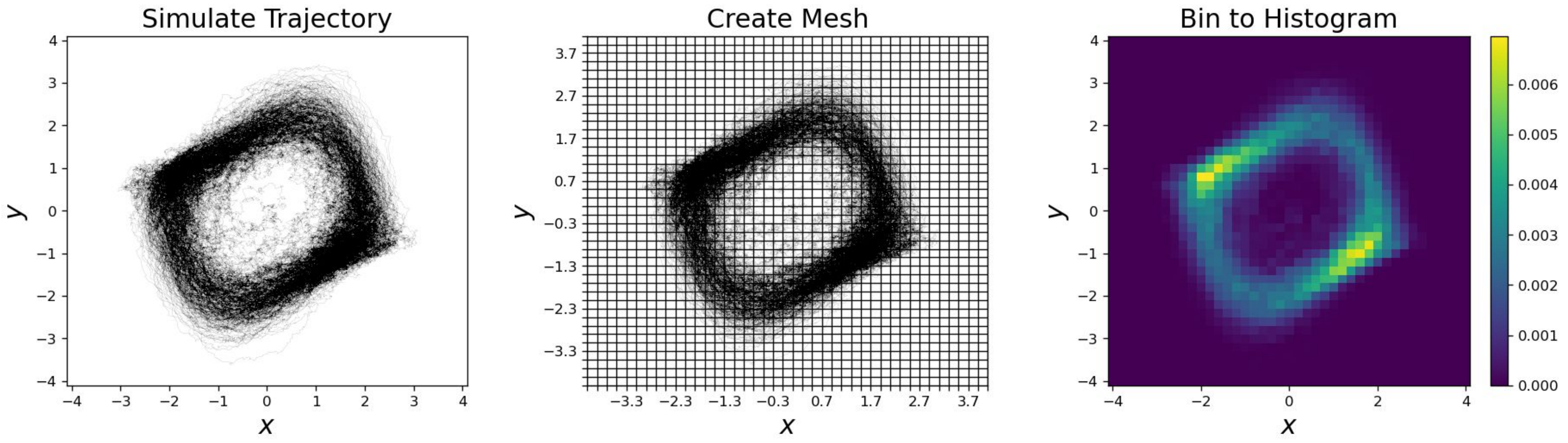


How do we computationally approximate a physical measure?

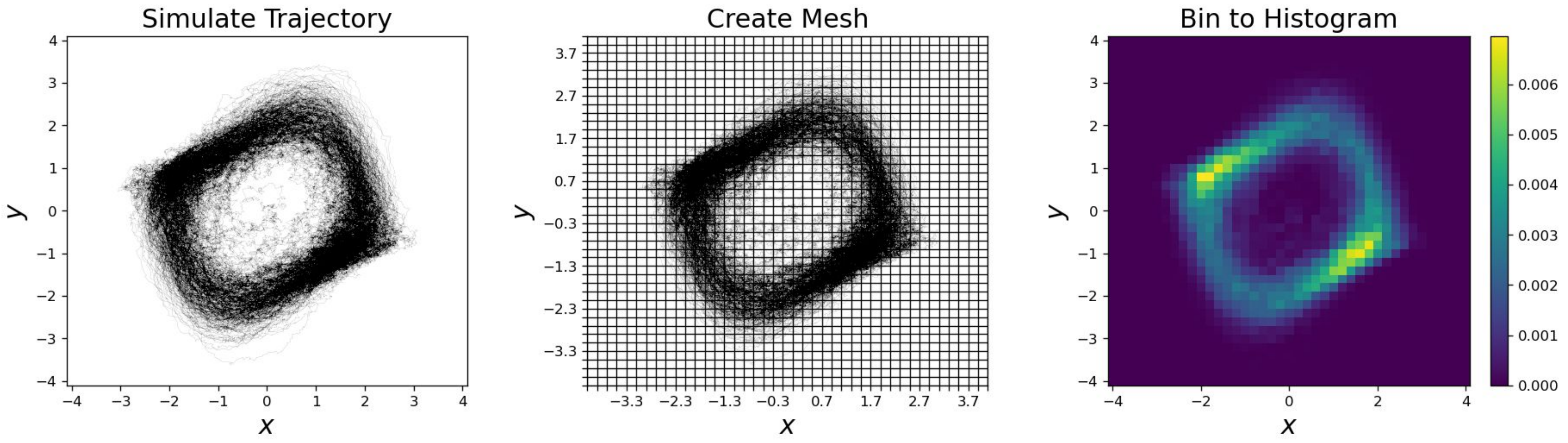




How do we computationally approximate a physical measure?



How do we computationally approximate a physical measure?



Note: this procedure does not use the sampling times of observations.

What is the goal?

Invert the mapping $v \mapsto \mu.$

If we know a physical invariant measure, can we infer the velocity field that produced it?

What is the goal?

Invert the mapping $v \mapsto \mu.$

If we know a physical invariant measure, can we infer the velocity field that produced it?

1.) Existence: ✓

2.) Uniqueness: ✗

3.) Stability: ✗

What is the goal?

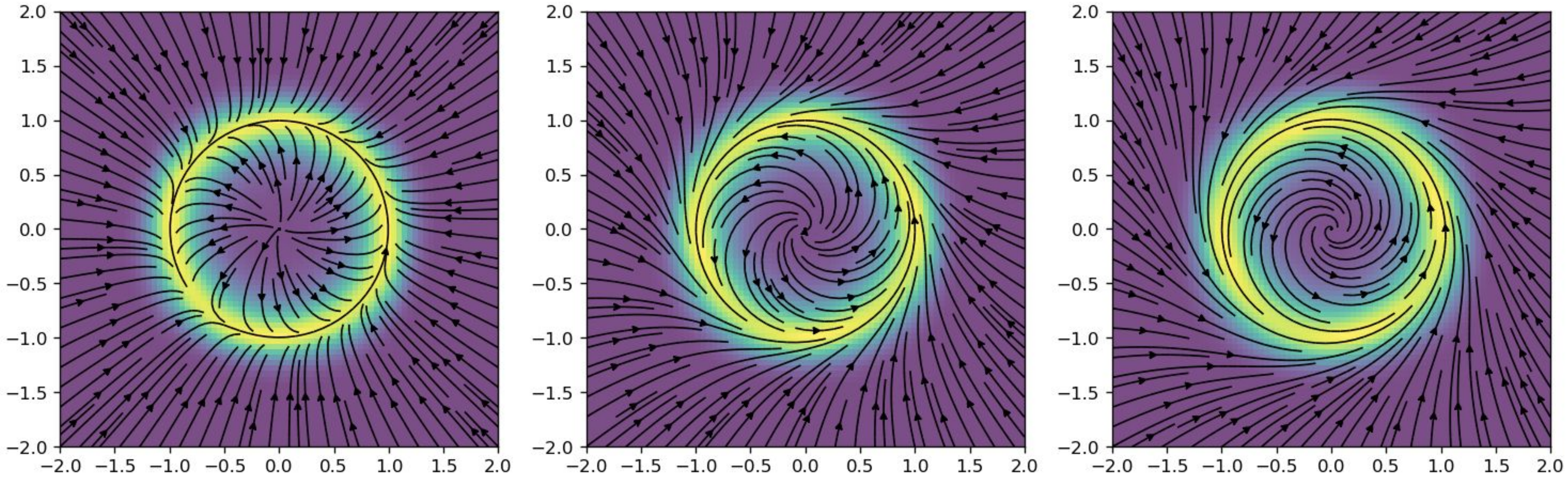
Invert the mapping $v \mapsto \mu.$

If we know a physical invariant measure, can we infer the velocity field that produced it?

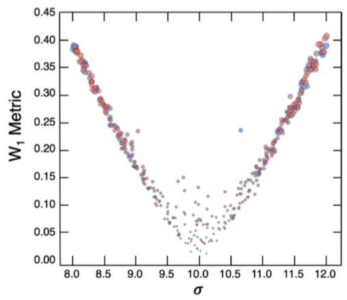
1.) Existence: ☑

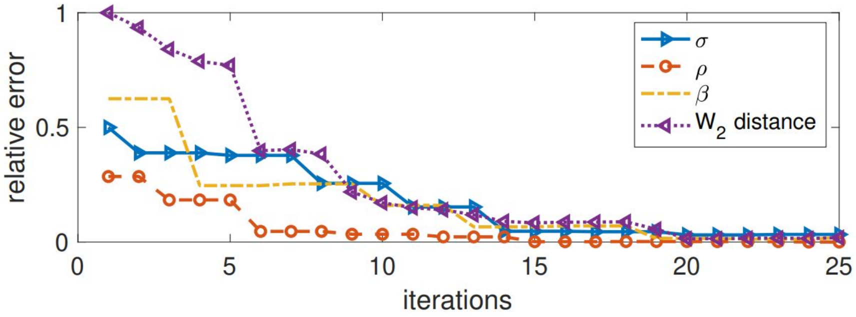
2.) Uniqueness: ✗

3.) Stability: ✗



Previous Work on Learning Dynamics via Invariant Measures

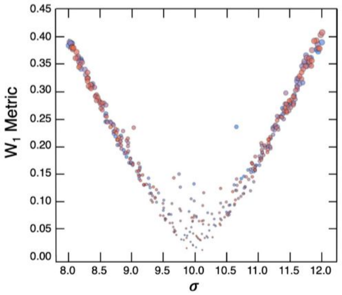


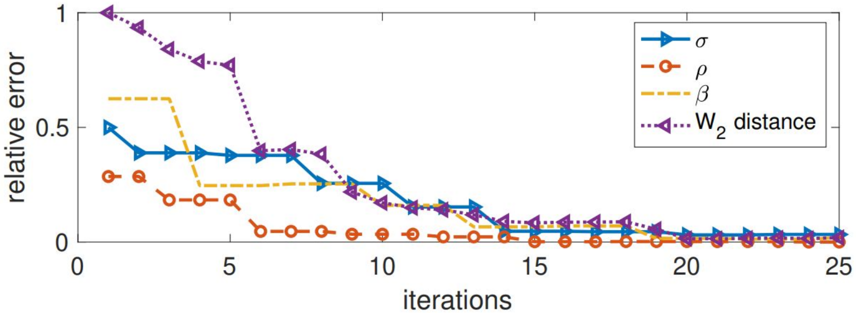


Yunan Yang, Levon Nurbekyan, Elisa Negrini, Robert Martin, and Mirjeta Pasha. Optimal transport for parameter identification of chaotic dynamics via invariant measures. *arXiv preprint arXiv:2104.15138*, 2021.

CM Greve, K Hara, RS Martin, DQ Eckhardt, and JW Koo. A data-driven approach to model calibration for nonlinear dynamical systems. *Journal of Applied physics*, 125(24):244901, 2019.

Previous Work on Learning Dynamics via Invariant Measures





Yunan Yang, Levon Nurbekyan, Elisa Negrini, Robert Martin, and Mirjeta Pasha. Optimal transport for parameter identification of chaotic dynamics via invariant measures. *arXiv preprint arXiv:2104.15138*, 2021.

CM Greve, K Hara, RS Martin, DQ Eckhardt, and JW Koo. A data-driven approach to model calibration for nonlinear dynamical systems. *Journal of Applied physics*, 125(24):244901, 2019.

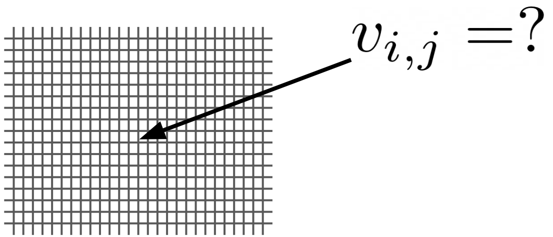
New contributions:

- Ability to model intrinsically noisy trajectories
- Large-scale parameter identification
- Learning dynamics in time-delay coordinates

Building a Forward Model

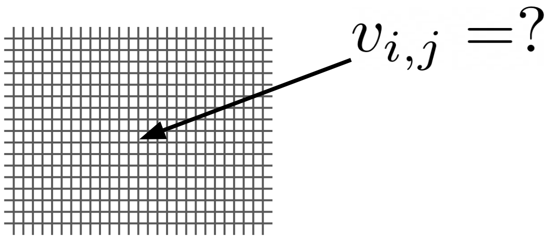
Reformulating the inversion as large-scale optimization

Discretize the velocity and "search" for a piecewise constant representation which inverts the map $v \mapsto \mu$.



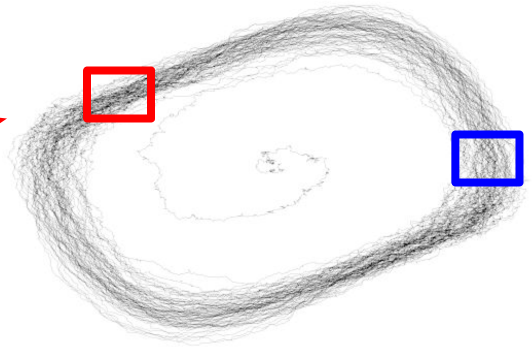
Reformulating the inversion as large-scale optimization

Discretize the velocity and “search” for a piecewise constant representation which inverts the map $v \mapsto \mu$.



It is difficult to compute the derivative of histogram bin counts with respect to the velocity.

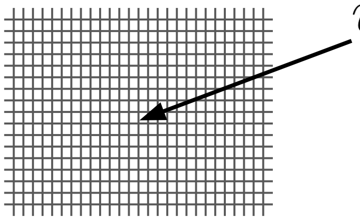
How does the number of particles here...



Depend on the velocity here?

Reformulating the inversion as large-scale optimization

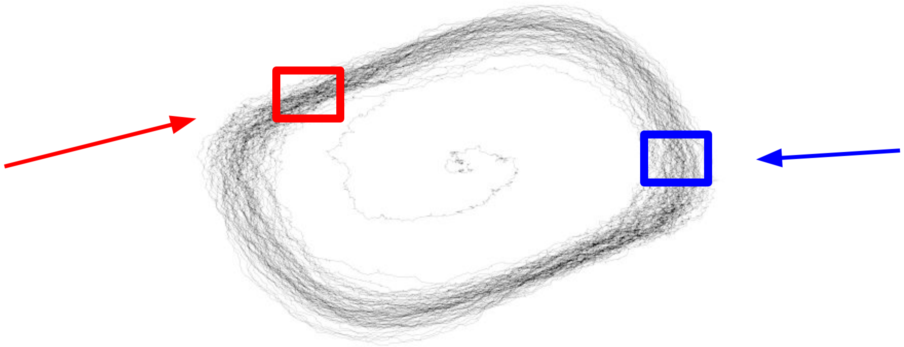
Discretize the velocity and “search” for a piecewise constant representation which inverts the map $v \mapsto \mu$.



$v_{i,j} =?$

It is difficult to compute the derivative of histogram bin counts with respect to the velocity.

How does the number of particles here...



Depend on the velocity here?

We need a forward model for which the mapping $v \mapsto \mu$ is easily differentiable!

A 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.

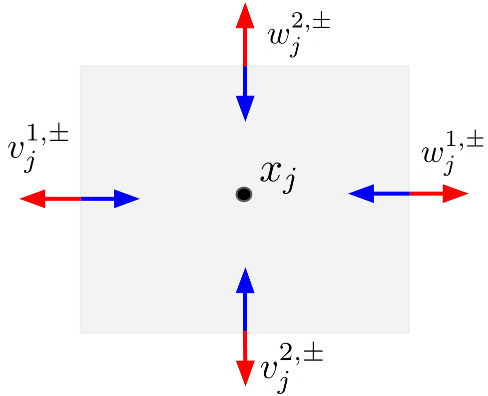
A PDE Forward Model

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

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.

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



Taking a closer look at the discretization…

$$K_i := \begin{array}{r} S_i \left\{ \vphantom{\begin{matrix} \ddots \\ \dfrac{D}{\Delta x_i} \\ \vdots \end{matrix}} \right. \\ \\ \\ \\ \\ \\ \\ \\ \end{array} \begin{bmatrix} \ddots & & & & \\ & -v_{j-1}^{i,-} + \dfrac{D}{\Delta x_i} & & & \\ \ddots & \vdots & & & \\ & v_{j-1}^{i,-} - w_{j-1}^{i,+} - \dfrac{2D}{\Delta x_i} & -v_j^{i,-} + \dfrac{D}{\Delta x_i} & & \\ \ddots & \vdots & \vdots & & \\ & w_{j-1}^{i,+} + \dfrac{D}{\Delta x_i} & v_j^{i,-} - w_j^{i,+} - \dfrac{2D}{\Delta x_i} & -v_{j+1}^{i,-} + \dfrac{D}{\Delta x_i} & \\ & & \vdots & \vdots & \ddots \\ & & w_j^{i,+} + \dfrac{D}{\Delta x_i} & v_{j+1}^{i,-} - w_{j+1}^{i,+} - 2\dfrac{D}{\Delta x_i} & \\ & & & \vdots & \ddots \\ & & & w_{j+1}^{i,+} + \dfrac{D}{\Delta x_i} & \\ & & & & \ddots \end{bmatrix} \in \mathbb{R}^{N \times N}.$$

Finding the Markov Chain's Stationary Distribution

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 Stationary Distribution

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 Stationary Distribution

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 Stationary Distribution

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.

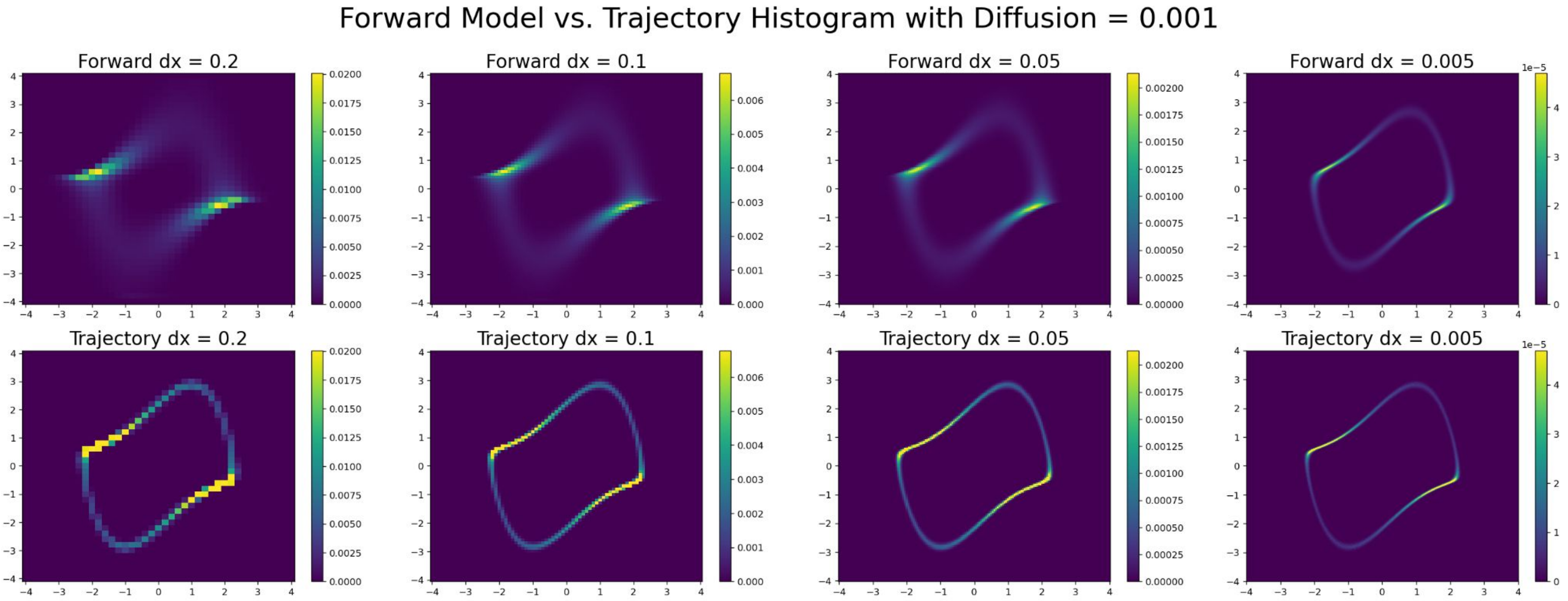
Solution: Teleportation regularization from Google's Pagerank algorithm.

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

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



Selecting an 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

Using the Adjoint State Method

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

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

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

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

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

Using 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

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

Using 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

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

The first term comes from the adjoint state method and the second is easy to compute when the functional form of the paramaterization is known.

Velocity Parameterization

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

The first term comes from the adjoint state method and the second is easy to compute when the functional form of the paramaterization is known.

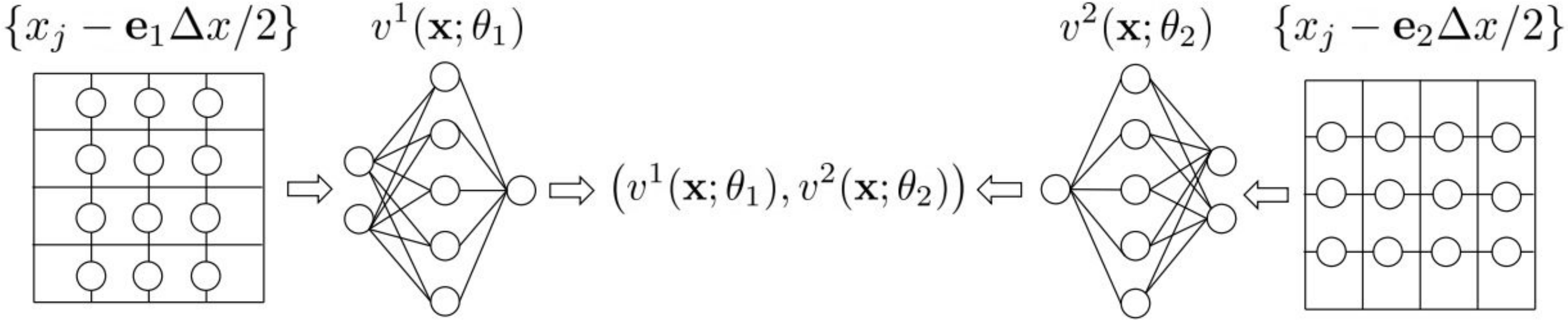
We tested three paramaterizations: piecewise constant, global polynomial, and neural network.

Velocity Parameterization

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

The first term comes from the adjoint state method and the second is easy to compute when the functional form of the paramaterization is known.

We tested three paramaterizations: piecewise constant, global polynomial, and neural network.



The Optimization Framework

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

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

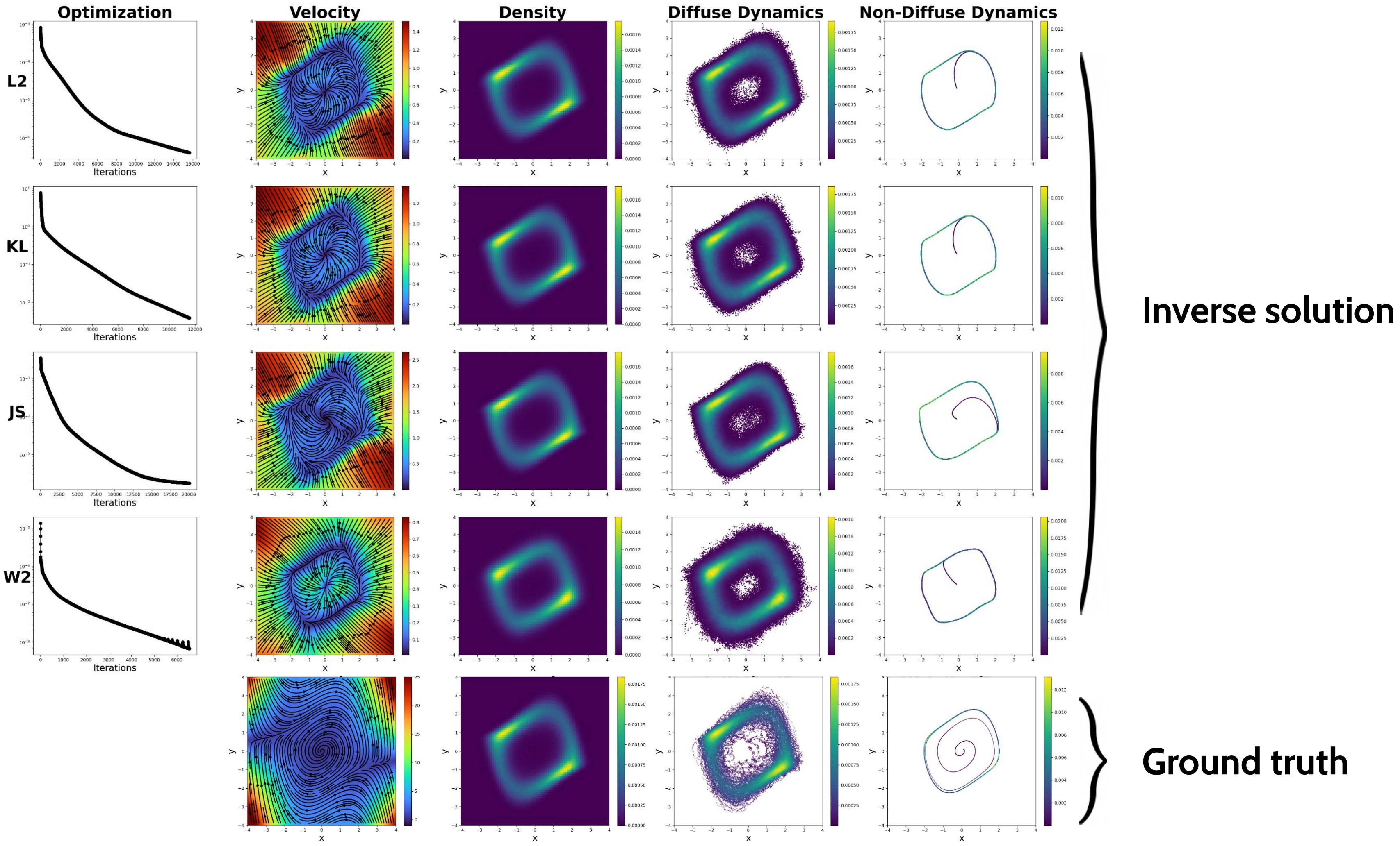
3.) Compute the Frechet derivative $$\phi = \frac{\partial \mathcal{J}}{\partial \rho}$$

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

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

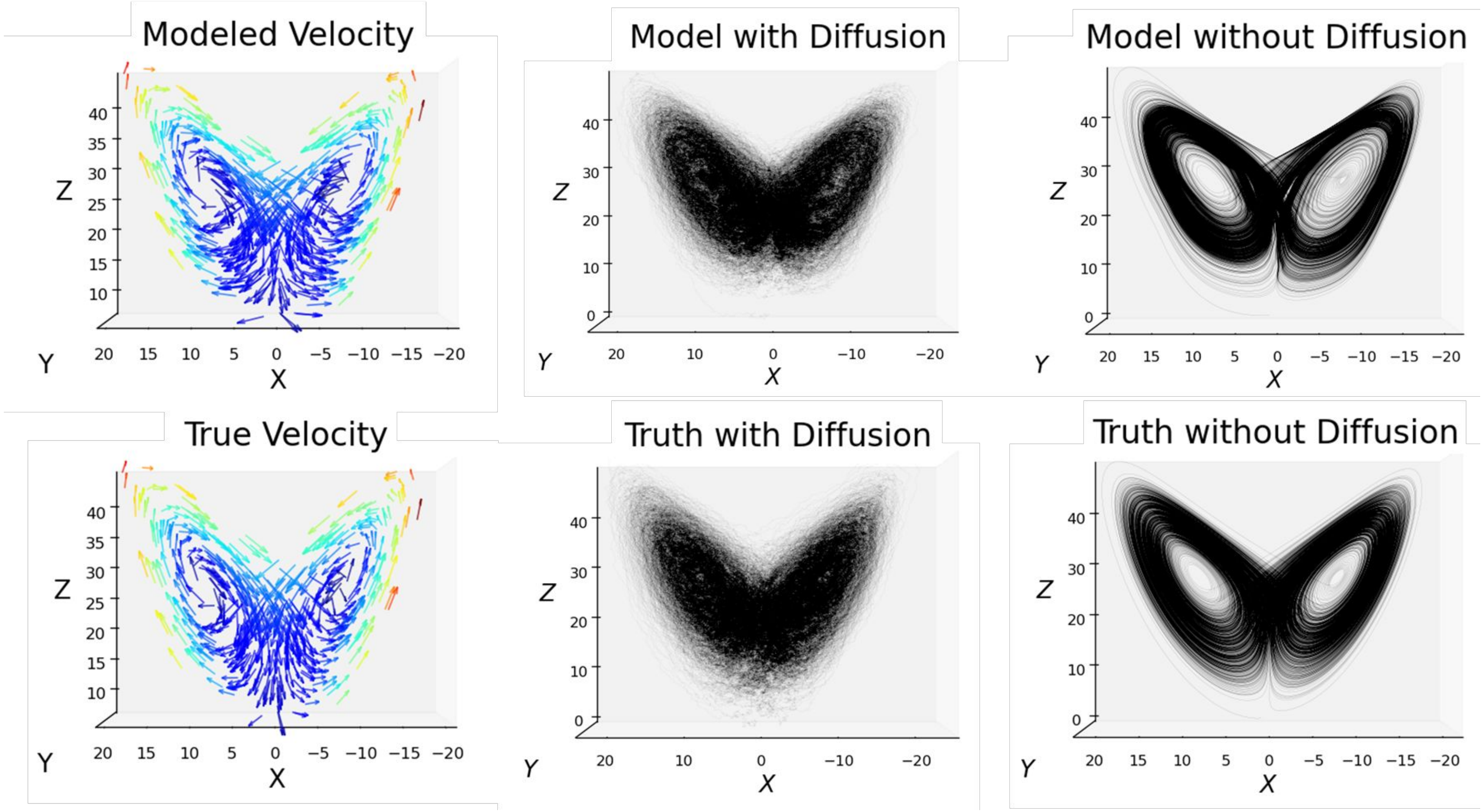
6.) Descend Adam, L-BFGS-B, CG, etc.

Numerical Results



Optimization
Velocity
Density
Diffuse Dynamics
Non-Diffuse Dynamics
L2
KL
JS
W2
Iterations
Inverse solution
Ground truth

Lorenz-63 System - Inverting V1



What if we only have partial observations?

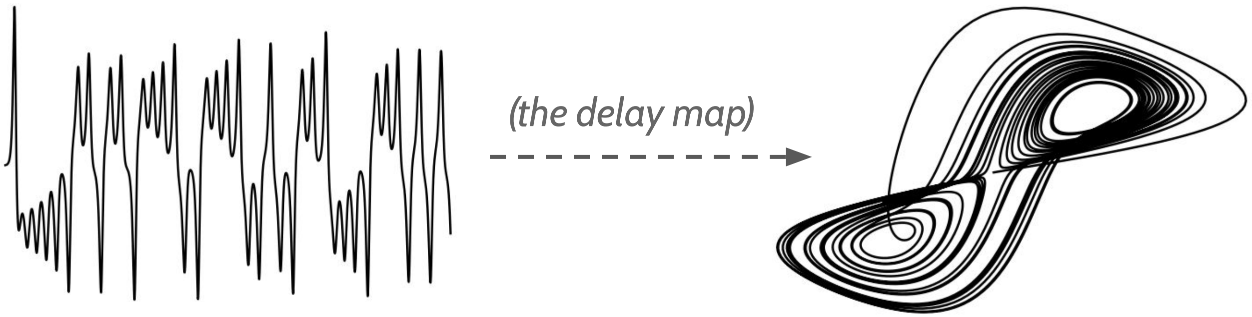
$$\underbrace{\dot{x} = v(x)}_{\text{full dynamics}} \qquad \underbrace{y : \mathcal{M} \to \mathbb{R}}_{\text{observation function}} \qquad \underbrace{\{y(x(t_i))\}_{i=1}^{N}}_{\text{available data}}$$

What if we only have partial observations?

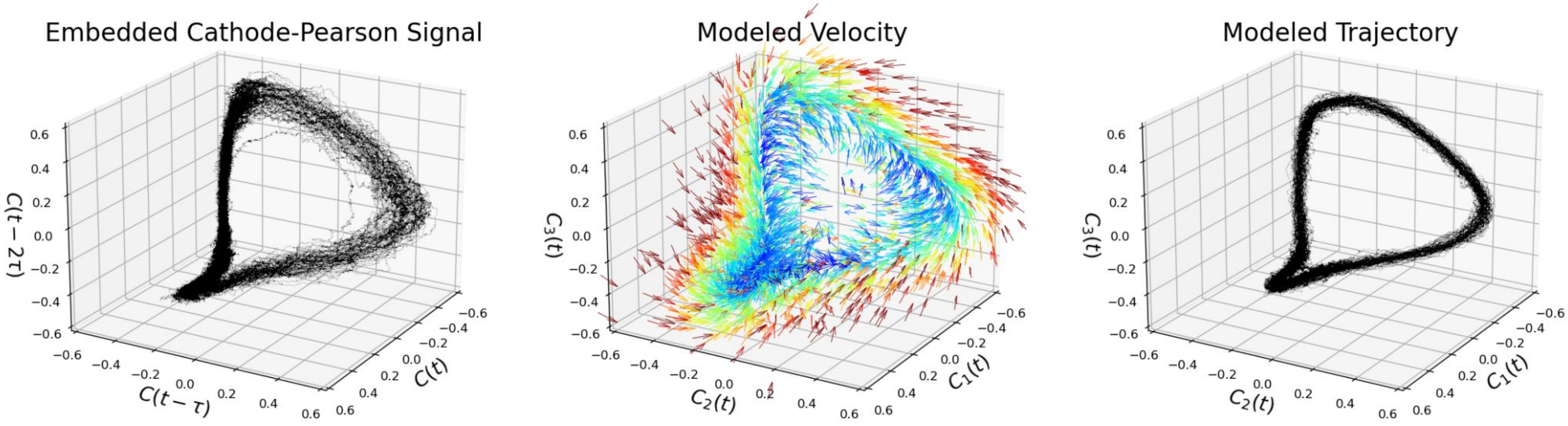
$$\underbrace{\dot{x} = v(x)}_{\text{full dynamics}} \qquad \underbrace{y : \mathcal{M} \to \mathbb{R}}_{\text{observation function}} \qquad \underbrace{\{y(x(t_i))\}_{i=1}^{N}}_{\text{available data}}$$

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

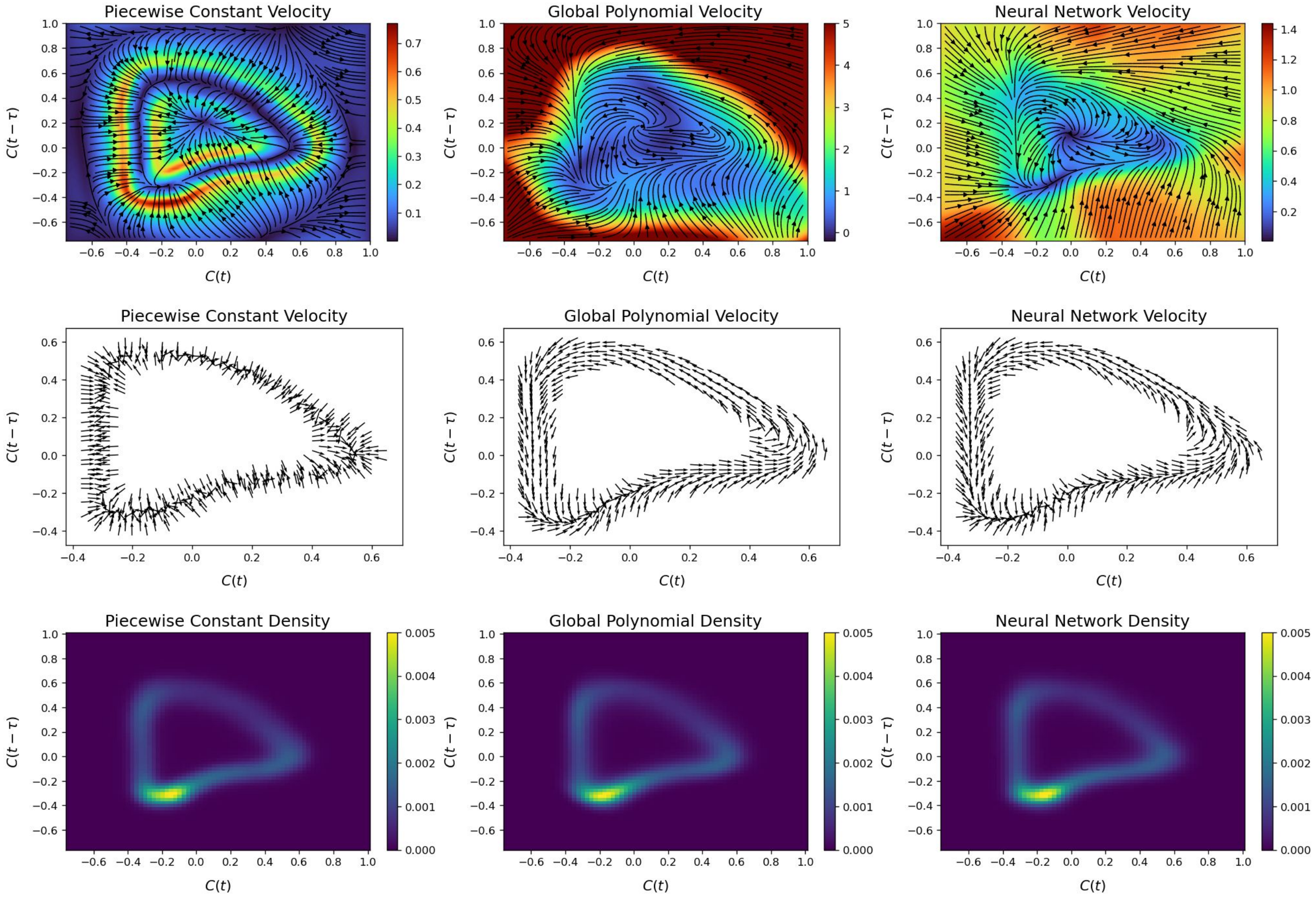
$$\exists \underbrace{\Phi : \mathcal{M} \to \mathbb{R}^d}_{\text{diffeomorphism}} \text{ with } \Phi(x(t)) = (y(t), y(t-\tau), \ldots, y(t-2d\tau))$$



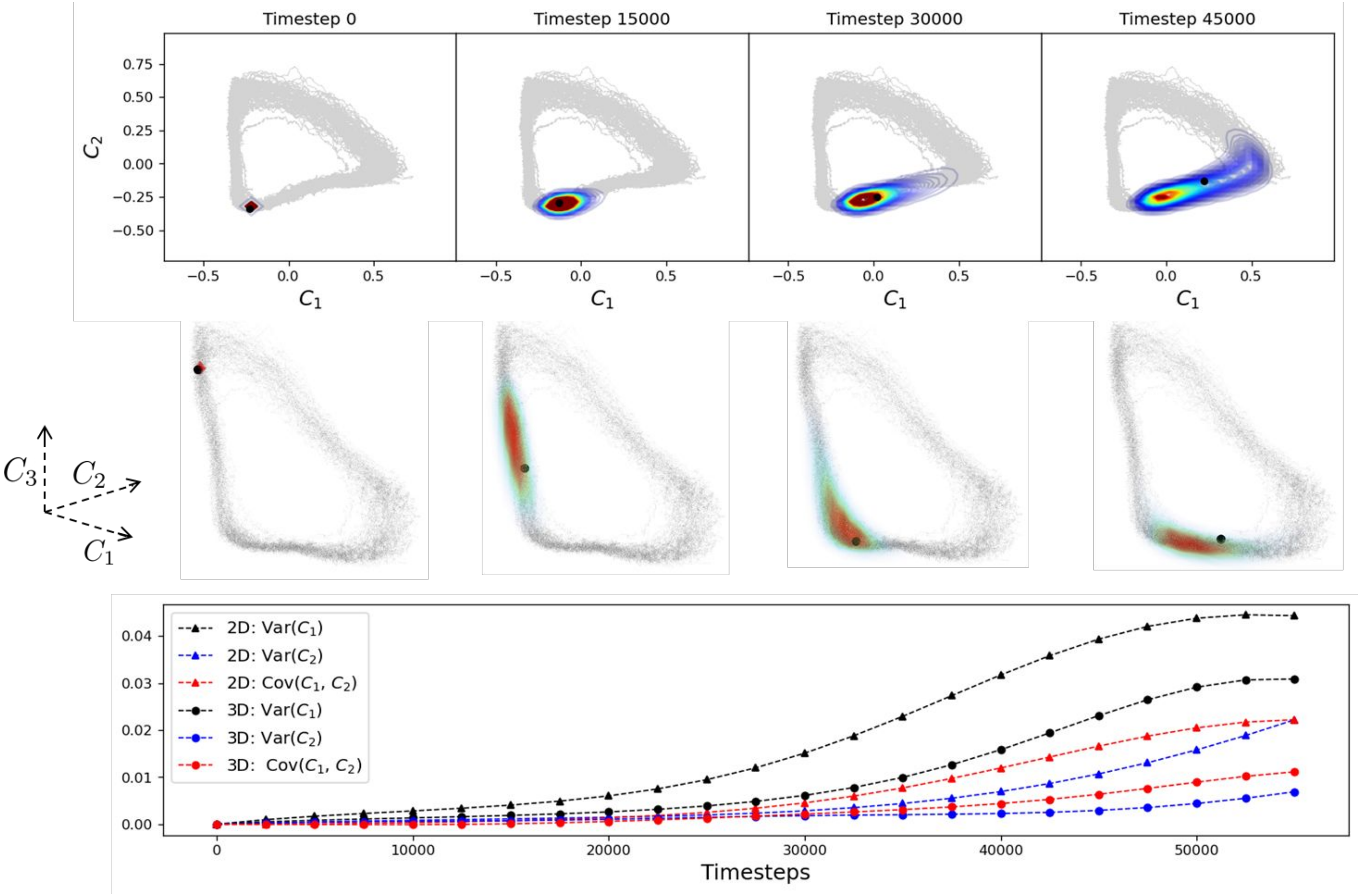
Application to a Hall-Effect Thruster (HET)



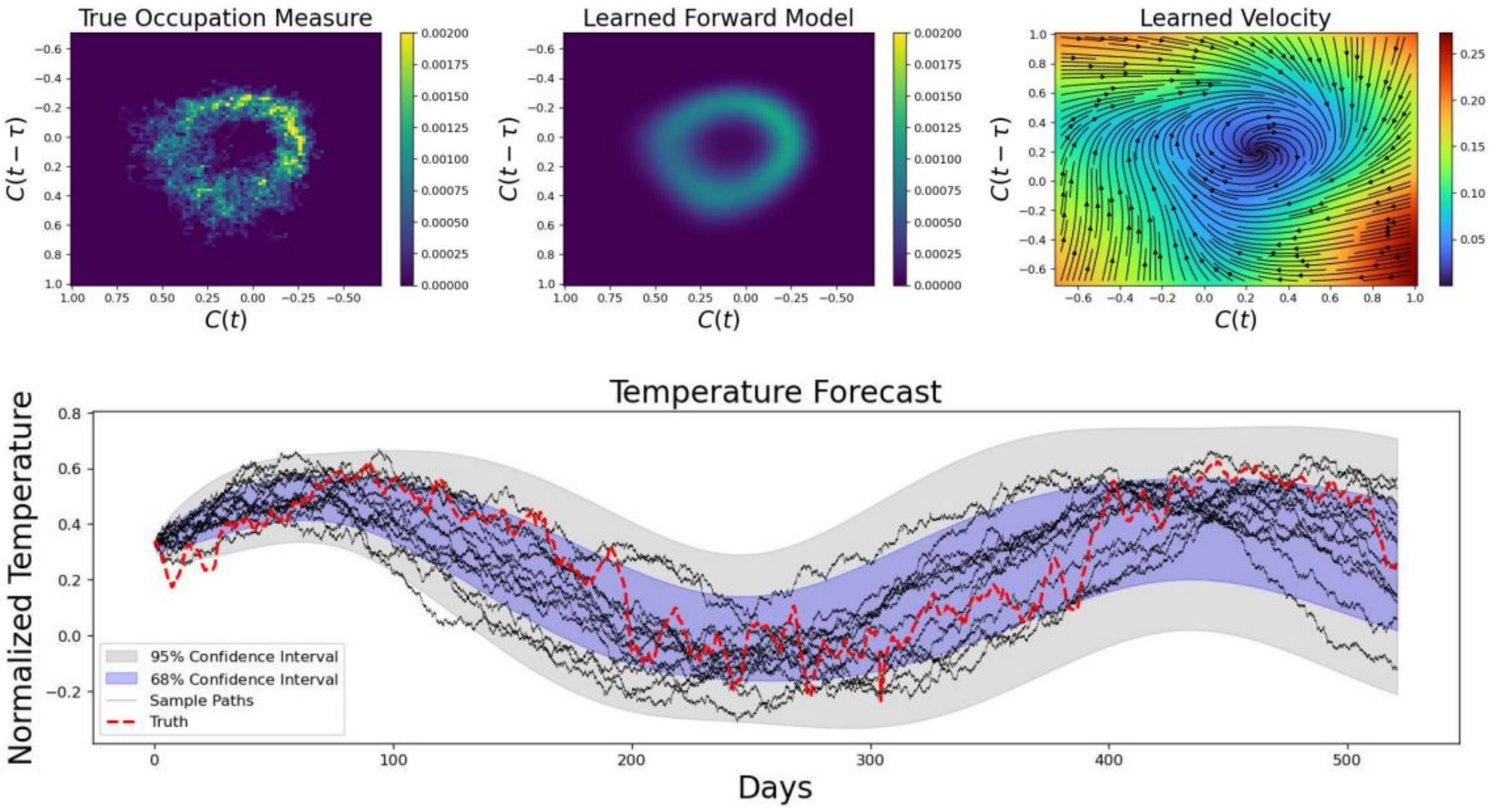
Varying the Paramaterization



Quantifying Model Uncertainty



Temperature Prediction with Uncertainty Quantification



Future Directions

- Dimension-free and mesh-free approaches
- Unstructured Mesh
- Higher order finite volume method
- Study inverse problem regularity
- The case of multiple attractors
- Learning an anisotropic diffusion

Thank you!

Questions?