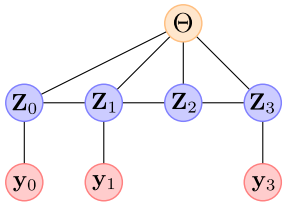
Data assimilation via low-rank couplings

D. Bigoni (**dabi@mit.edu**), R. Baptista, A. Spantini, Y. Marzouk
Massachusetts Institute of Technology

ICIAM
Valencia, Spain – 16/07/2019

Sequential data assimilation

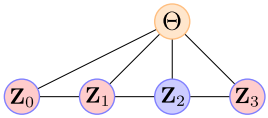


$$\pi\left(\Theta, \mathbf{Z}_\Lambda | \mathbf{y}_\Xi\right) \propto \mathcal{L}\left(\mathbf{y}_\Xi | \Theta, \mathbf{Z}_\Lambda\right) \pi\left(\Theta, \mathbf{Z}_\Lambda\right)$$

$$\mathcal{L}\left(\mathbf{y}_\Xi | \Theta, \mathbf{Z}_\Lambda\right) = \prod_{k \in \Xi} \mathcal{L}\left(\mathbf{y}_k | \Theta, \mathbf{Z}_k\right)$$

$$\pi\left(\Theta, \mathbf{Z}_\Lambda\right) = \pi\left(\Theta\right) \pi\left(\mathbf{Z}_0 | \Theta\right) \prod_{k \in \Lambda} \pi\left(\mathbf{Z}_k | \mathbf{Z}_{k-1}, \Theta\right)$$

Sequential data assimilation

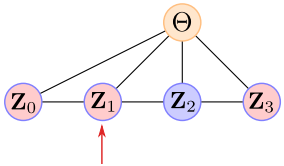


$$\pi\left(\Theta, \mathbf{Z}_\Lambda | \mathbf{y}_\Xi\right) \propto \mathcal{L}\left(\mathbf{y}_\Xi | \Theta, \mathbf{Z}_\Lambda\right) \pi\left(\Theta, \mathbf{Z}_\Lambda\right)$$

$$\mathcal{L}\left(\mathbf{y}_\Xi | \Theta, \mathbf{Z}_\Lambda\right) = \prod_{k \in \Xi} \mathcal{L}\left(\mathbf{y}_k | \Theta, \mathbf{Z}_k\right)$$

$$\pi\left(\Theta, \mathbf{Z}_\Lambda\right) = \pi\left(\Theta\right) \pi\left(\mathbf{Z}_0 | \Theta\right) \prod_{k \in \Lambda} \pi\left(\mathbf{Z}_k | \mathbf{Z}_{k-1}, \Theta\right)$$

Sequential data assimilation

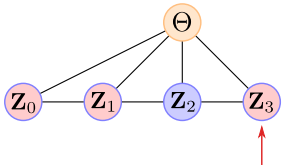


$$\pi\left(\mathbf{Z}_k|\mathbf{y}_\Xi\right) \propto \int \mathcal{L}\left(\mathbf{y}_\Xi|\Theta,\mathbf{Z}_\Lambda\right)\pi\left(\Theta,\mathbf{Z}_\Lambda\right)d\Theta\, d\mathbf{Z}_{j\neq k}$$

$$\mathcal{L}\left(\mathbf{y}_\Xi|\Theta,\mathbf{Z}_\Lambda\right) = \prod_{k\in\Xi}\mathcal{L}\left(\mathbf{y}_k|\Theta,\mathbf{Z}_k\right)$$

$$\pi\left(\Theta,\mathbf{Z}_\Lambda\right) = \pi\left(\Theta\right)\pi\left(\mathbf{Z}_0|\Theta\right)\prod_{k\in\Lambda}\pi\left(\mathbf{Z}_k|\mathbf{Z}_{k-1},\Theta\right)$$

Sequential data assimilation



$$\pi\left(\mathbf{Z}_k|\mathbf{y}_{j\leq k}\right) \propto \int \mathcal{L}\left(\mathbf{y}_{j\leq k}|\Theta, \mathbf{Z}_{j\leq k}\right) \pi\left(\Theta, \mathbf{Z}_{j\leq k}\right) d\Theta \, d\mathbf{Z}_{j<k}$$

$$\mathcal{L}\left(\mathbf{y}_\Xi|\Theta, \mathbf{Z}_\Lambda\right) = \prod_{k\in\Xi} \mathcal{L}\left(\mathbf{y}_k|\Theta, \mathbf{Z}_k\right)$$

$$\pi\left(\Theta, \mathbf{Z}_\Lambda\right) = \pi\left(\Theta\right) \pi\left(\mathbf{Z}_0|\Theta\right) \prod_{k\in\Lambda} \pi\left(\mathbf{Z}_k|\mathbf{Z}_{k-1}, \Theta\right)$$

Outline

Transport maps

Data assimilation via low-dimensional couplings

Low-rank structure in lag-1 updates

Results

Transport maps – Pullbacks [PB] and Pushforwards [PF]

- Distribution $\boldsymbol{\nu}_\rho$ with density $\rho : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$
- Distribution $\boldsymbol{\nu}_\pi$ with density $\pi : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$
- For $T : \mathbb{R}^d \to \mathbb{R}^d$ we define

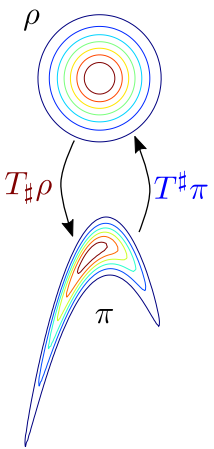
  PF $\quad T_\sharp \rho = \rho \circ T^{-1} |\nabla T^{-1}|$

  PB $\quad T^\sharp \pi = \pi \circ T |\nabla T|$

- We want T such that

  PF $\quad T_\sharp \rho = \pi$

  PB $\quad T^\sharp \pi = \rho$



Transport maps – Pullbacks [PB] and Pushforwards [PF]

- Distribution $\boldsymbol{\nu}_\rho$ with density $\rho : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$
- Distribution $\boldsymbol{\nu}_\pi$ with density $\pi : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$
- For $T : \mathbb{R}^d \to \mathbb{R}^d$ we define

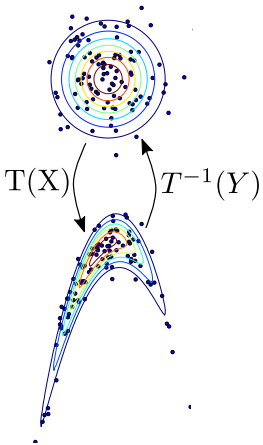
  PF $\quad T_\sharp \rho = \rho \circ T^{-1} |\nabla T^{-1}|$

  PB $\quad T^\sharp \pi = \pi \circ T |\nabla T|$

- We want T such that

  PF For $X \sim \boldsymbol{\nu}_\rho$, $T(X) \sim \boldsymbol{\nu}_\pi$

  PB For $Y \sim \boldsymbol{\nu}_\pi$, $T^{-1}(Y) \sim \boldsymbol{\nu}_\rho$



Transport maps – Pullbacks [PB] and Pushforwards [PF]

- Distribution $\boldsymbol{\nu}_\rho$ with density $\rho : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$
- Distribution $\boldsymbol{\nu}_\pi$ with density $\pi : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$
- For $T : \mathbb{R}^d \to \mathbb{R}^d$ we define

  PF $\quad T_\sharp \rho = \rho \circ T^{-1} |\nabla T^{-1}|$

  PB $\quad T^\sharp \pi = \pi \circ T |\nabla T|$

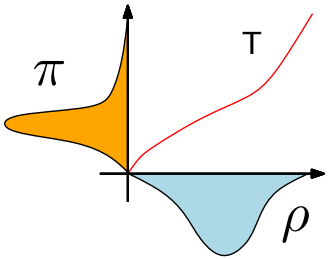
- We want T such that

  PF For $X \sim \boldsymbol{\nu}_\rho$, $T(X) \sim \boldsymbol{\nu}_\pi$

  PB For $Y \sim \boldsymbol{\nu}_\pi$, $T^{-1}(Y) \sim \boldsymbol{\nu}_\rho$

Knothe-Rosenblatt rearrangement

$\forall\, \boldsymbol{\nu}_\rho, \boldsymbol{\nu}_\pi$ Lebesgue absolutely continuous
$\exists$ a **triangular monotone** map s.t. $T_\sharp \rho = \pi$



$$T(\mathbf{x}) = \begin{bmatrix} T^{(1)}(x_1) \\ T^{(2)}(x_1, x_2) \\ \vdots \\ T^{(d)}(x_1, \ldots, x_d) \end{bmatrix}$$

Triangular monotone maps

$$\mathcal{T}_> = \left\{ T : \mathbb{R}^d \to \mathbb{R}^d : \overbrace{[T(\mathbf{x})]_k = T^{(k)}(x_1, \dots, x_k)}^{\text{triangular}} \text{ and } \overbrace{\partial_{x_k} T^{(k)} > 0}^{\text{monotone}} \right\}$$

Triangular monotone maps

$$\mathcal{T}_> = \left\{ T : \mathbb{R}^d \to \mathbb{R}^d : \overbrace{[T(\mathbf{x})]_k = T^{(k)}(x_1, \dots, x_k)}^{\text{triangular}} \text{ and } \overbrace{\partial_{x_k} T^{(k)} > 0}^{\text{monotone}} \right\}$$

Integrated squared representation – $\varepsilon > 0$

$$T^{(k)}(x_{1:k}) = c_k(x_{1:k-1}) + \int_0^{x_k} \Big(h_k(x_{1:k-1}, t) \Big)^2 + \varepsilon \, dt$$

Triangular monotone maps

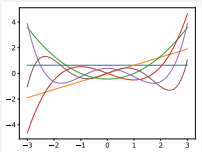
$$\boxed{\mathcal{T}_>^n} = \left\{ T : \mathbb{R}^d \to \mathbb{R}^d : \overbrace{[T(\mathbf{x})]_k = T^{(k)}(x_1, \dots, x_k)}^{\text{triangular}} \text{ and } \overbrace{\partial_{x_k} T^{(k)} > 0}^{\text{monotone}} \right\}$$

Integrated squared representation – $\varepsilon > 0$

$$T^{(k)}(x_{1:k}) = \boxed{c_k(x_{1:k-1})} + \int_0^{x_k} \left(\boxed{h_k(x_{1:k-1}, t)} \right)^2 + \varepsilon \, dt$$

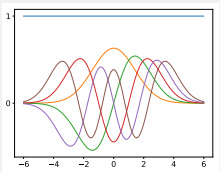
Constant part

$c_k(x_{1:k-1}) = \sum_{\mathbf{i} \in \mathcal{I}_k} \mathbf{c_i} \Phi_{\mathbf{i}}(x_{1:k-1})$



Squared part

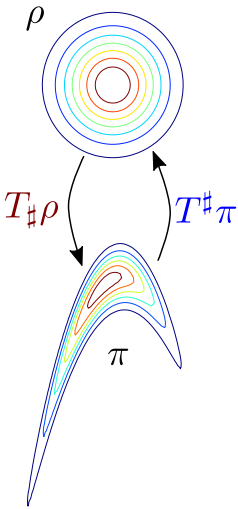
$h_k(x_{1:k-1}, t) = \sum_{\mathbf{j} \in \mathcal{J}_k} \mathbf{h_j} \Psi_{\mathbf{j}}(x_{1:k-1}, t)$



Knothe-Rosenblatt rearrangement

$\forall\ \boldsymbol{\nu}_\rho, \boldsymbol{\nu}_\pi$ Lebesgue absolutely continuous
$\exists$ a **triangular monotone** map s.t. $T_\sharp \rho = \pi$

How to find the map $T \in \mathcal{T}_>$ such that $T_\sharp \rho = \pi$?



Minimize KL-divergence to find optimal map [ElMoselhy et al. 2012]

$$\widehat{T} = \underset{T \in \mathcal{T}_>}{\arg\min}\, D_{\mathrm{KL}}(T_\sharp \boldsymbol{\nu}_\rho \| \boldsymbol{\nu}_\pi) = \underset{T \in \mathcal{T}_>}{\arg\min}\, \mathbb{E}_\rho \left[\log \frac{\rho}{T^\sharp \pi} \right]$$

Minimize KL-divergence to find optimal map [ElMoselhy et al. 2012]

$$\widehat{T} = \underset{T \in \mathcal{T}_>}{\arg\min}\, D_{\mathrm{KL}}(T_\sharp \boldsymbol{\nu}_\rho \| \boldsymbol{\nu}_\pi) = \underset{T \in \mathcal{T}_>}{\arg\min}\, \mathbb{E}_\rho \left[\log \frac{\rho}{T^\sharp \pi} \right]$$

In particular we will:

- Discretize $\mathbb{E}_\rho[f(\mathbf{x})] \approx \sum_{i=1}^m f(\mathbf{x}_i)\, \mathbf{w}_i$
- Work on the n-dimensional space $\mathcal{T}_>^n$ of maps $T[\mathbf{a}]$ with parameters $\mathbf{a} \in \mathbb{R}^n$

Minimize KL-divergence to find optimal map [ElMoselhy et al. 2012]

$$\widehat{T} = \underset{T \in \mathcal{T}_>}{\arg\min}\, D_{\mathrm{KL}}(T_\sharp \boldsymbol{\nu}_\rho \| \boldsymbol{\nu}_\pi) = \underset{T \in \mathcal{T}_>}{\arg\min}\, \mathbb{E}_\rho \left[\log \frac{\rho}{T^\sharp \pi} \right]$$

In particular we will:

- Discretize $\mathbb{E}_\rho[f(\mathbf{x})] \approx \sum_{i=1}^m f(\mathbf{x}_i)\, \mathbf{w}_i$
- Work on the n-dimensional space $\mathcal{T}_>^n$ of maps $T[\mathbf{a}]$ with parameters $\mathbf{a} \in \mathbb{R}^n$

$$\widetilde{\mathbf{a}} = \underset{\mathbf{a} \in \mathbb{R}^n}{\arg\min} \sum_{i=1}^m \left[-\log T[\mathbf{a}]^\sharp \pi(\mathbf{x}_i) \right] \mathbf{w}_i$$

Minimize KL-divergence to find optimal map [ElMoselhy et al. 2012]

$$\widehat{T} = \underset{T \in \mathcal{T}_>}{\arg\min}\, D_{\mathrm{KL}}(T_\sharp \boldsymbol{\nu}_\rho \| \boldsymbol{\nu}_\pi) = \underset{T \in \mathcal{T}_>}{\arg\min}\, \mathbb{E}_\rho \left[\log \frac{\rho}{T^\sharp \pi} \right]$$

+ **Gradient-based unconstrained optimization** if gradients are available
+ We can **explore π in parallel**
+ We can **generate i.i.d. samples** from $\widehat{T}_\sharp \boldsymbol{\nu}_\rho = \boldsymbol{\nu}_\pi$ **in parallel**

Minimize KL-divergence to find optimal map [ElMoselhy et al. 2012]

$$\widehat{T} = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} D_{\mathrm{KL}}(T_\sharp \boldsymbol{\nu}_\rho \| \boldsymbol{\nu}_\pi) = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} \mathbb{E}_\rho \left[\log \frac{\rho}{T^\sharp \pi} \right]$$

+ **Gradient-based unconstrained optimization** if gradients are available
+ We can **explore π in parallel**
+ We can **generate i.i.d. samples** from $\widehat{T}_\sharp \boldsymbol{\nu}_\rho = \boldsymbol{\nu}_\pi$ **in parallel**

We are working on $\mathcal{T}_>^n \subset \mathcal{T}_>$, so
how can we **evaluate the quality of the approximation**?

Convergence criterion – Variance diagnostic

$$\widehat{T} = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} D_{\mathrm{KL}}(T_\sharp \boldsymbol{\nu}_\rho \| \boldsymbol{\nu}_\pi) = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} \mathbb{E}_\rho \left[\log \frac{\rho}{T^\sharp \widetilde{\pi}} \right] + \log \int \widetilde{\pi}$$

$$\text{Optimal } \widehat{T} \in \mathcal{T}_> \text{ and } \int \widetilde{\pi} = 1 \quad \Rightarrow \quad \mathbb{E}_\rho \left[\log \frac{\rho}{(\widehat{T})^\sharp \widetilde{\pi}} \right] = 0$$

$$\text{But, optimal } \widetilde{T} \in \mathcal{T}_>^n \text{ or } \int \widetilde{\pi} \neq 1 \quad \Rightarrow \quad \mathbb{E}_\rho \left[\log \frac{\rho}{(\widetilde{T})^\sharp \widetilde{\pi}} \right] \neq 0$$

Convergence criterion – Variance diagnostic

$$\widehat{T} = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} D_{\mathrm{KL}}(T_\sharp \boldsymbol{\nu}_\rho \| \boldsymbol{\nu}_\pi) = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} \mathbb{E}_\rho \left[\log \frac{\rho}{T^\sharp \widetilde{\pi}} \right] + \log \int \widetilde{\pi}$$

$$\text{Optimal } \widehat{T} \in \mathcal{T}_> \text{ and } \int \widetilde{\pi} = 1 \quad \Rightarrow \quad \mathbb{E}_\rho \left[\log \frac{\rho}{(\widehat{T})^\sharp \widetilde{\pi}} \right] = 0$$

$$\text{But, optimal } \widetilde{T} \in \mathcal{T}_>^n \text{ or } \int \widetilde{\pi} \neq 1 \quad \Rightarrow \quad \mathbb{E}_\rho \left[\log \frac{\rho}{(\widetilde{T})^\sharp \widetilde{\pi}} \right] \neq 0$$

$$D_{\mathrm{KL}}(T_\sharp \boldsymbol{\nu}_\rho \| \boldsymbol{\nu}_\pi) \;\approx\; \tfrac{1}{2} \mathbb{V} \left[\log \tfrac{\rho}{T^\sharp \widetilde{\pi}} \right] \quad \text{as} \quad T \;\to\; \widehat{T}$$

Pros & cons

$$\widehat{T} = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} D_{\mathrm{KL}}(T_\sharp \rho \| \pi) = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} \mathbb{E}_\rho \left[\log \frac{\rho}{T^\sharp \pi} \right]$$

+ **Gradient-based unconstrained optimization** if gradients are available
+ We can **explore π in parallel**
+ We can **generate i.i.d. samples** from $\widehat{T}_\sharp \boldsymbol{\nu}_\rho = \boldsymbol{\nu}_\pi$ **in parallel**
+ We can **assess convergence**!

Pros & cons

$$\widehat{T} = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} D_{\mathrm{KL}}(T_\sharp \rho \| \pi) = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} \mathbb{E}_\rho \left[\log \frac{\rho}{T^\sharp \pi} \right]$$

+ **Gradient-based unconstrained optimization** if gradients are available
+ We can **explore π in parallel**
+ We can **generate i.i.d. samples** from $\widehat{T}_\sharp \boldsymbol{\nu}_\rho = \boldsymbol{\nu}_\pi$ **in parallel**
+ We can **assess convergence**!

$\widetilde{T}_\sharp \rho$ is a **biased** approximation of π.
How to reduce such bias?

Use $\widehat{T}$ as a preconditioner for ... Markov Chain Monte Carlo

1. Generate the Markov chain $\{\mathbf{x}_i\}$ with invariant distribution $\widetilde{T}^\sharp \pi$ (use your favorite MCMC method/proposals)
2. The Markov chain $\{\widetilde{T}(\mathbf{x}_i)\}$ has invariant distribution π

Use $\widehat{T}$ as a preconditioner for ... Markov Chain Monte Carlo

1. Generate the Markov chain $\{\mathbf{x}_i\}$ with invariant distribution $\widetilde{T}^\sharp\pi$ (use your favorite MCMC method/proposals)
2. The Markov chain $\{\widetilde{T}(\mathbf{x}_i)\}$ has invariant distribution π

If $\widetilde{T}^\sharp\widetilde{\pi} \overset{\propto}{\approx} \rho$ then ρ is a **good proposal** for $\widetilde{T}^\sharp\widetilde{\pi}$.

Pros & cons

$$\widehat{T} = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} D_{\mathrm{KL}}(T_\sharp \rho \| \pi) = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} \mathbb{E}_\rho \left[\log \frac{\rho}{T^\sharp \pi} \right]$$

+ **Gradient-based unconstrained optimization** if gradients are available
+ We can **explore π in parallel**
+ We can **generate i.i.d. samples** from $\widehat{T}_\sharp \boldsymbol{\nu}_\rho = \boldsymbol{\nu}_\pi$ **in parallel**
+ We can **assess convergence**!
+ The map can be used as a **preconditioner**

Pros & cons

$$\widehat{T} = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} D_{\mathrm{KL}}(T_\sharp \rho \| \pi) = \operatorname*{arg\,min}_{T \in \mathcal{T}_>} \mathbb{E}_\rho \left[\log \frac{\rho}{T^\sharp \pi} \right]$$

+ **Gradient-based unconstrained optimization** if gradients are available
+ We can **explore π in parallel**
+ We can **generate i.i.d. samples** from $\widehat{T}_\sharp \boldsymbol{\nu}_\rho = \boldsymbol{\nu}_\pi$ **in parallel**
+ We can **assess convergence**!
+ The map can be used as a **preconditioner**
− We need to **approximate d functions of up to d variables!**

$$T(\mathbf{x}) = \begin{bmatrix} T^{(1)}(x_1) \\ T^{(2)}(x_1, x_2) \\ \vdots \\ T^{(d)}(x_1, \dots, x_d) \end{bmatrix}$$

Exploit source of low-dimensional structure

1. Smoothness and marginal independence
   - Ongoing work...
2. Conditional independence
   - Variational filtering/smoothing and parameter estimation

     [Spantini, B, Marzouk 2018; Houssineau, Jasra, Singh 2018]
   - Ensamble filtering and smoothing

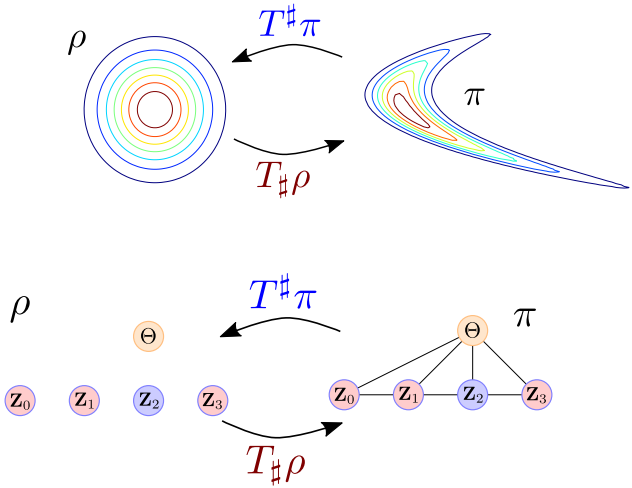
     [Spantini, Baptista, Marzouk 2019]
3. Multilevel/multifidelity structure

   [Parno, Moselhy, Marzouk 2018; Peherstorfer, Marzouk 2019]
4. Low-rank structure

   [B, Zahm, Spantini, Marzouk 2019]

Data assimilation via low-dimensional couplings

How to remove conditional dependencies?



Assume at step $k \in \Lambda$ the map $\mathfrak{M}^1_{k-1}(\mathbf{z})$ is available and

$$(\mathfrak{M}^1_{k-1})_\sharp \rho(\mathbf{z}_k) = \pi\left(\mathbf{z}_k | \mathbf{y}_{j\in\Xi,\, j\leq k}\right) .$$

Let $\mathfrak{M}_k(\mathbf{z}_k, \mathbf{z}_{k+1}) = \begin{bmatrix} \mathfrak{M}^0_k(\mathbf{z}_k, \mathbf{z}_{k+1}) \\ \mathfrak{M}^1_k(\mathbf{z}_{k+1}) \end{bmatrix}$ be such that

$$(\mathfrak{M}_k)_\sharp \rho(\mathbf{z}_k, \mathbf{z}_{k+1}) = \begin{bmatrix} \mathfrak{M}^1_{k-1}(\mathbf{z}_k) \\ \mathbf{z}_{k+1} \end{bmatrix}^\sharp \pi(\mathbf{z}_k, \mathbf{z}_{k+1} | \mathbf{y}_{j\leq k+1}) .$$

Then the following holds true [Spantini, B, Marzouk 2018]:

Filtering: $(\mathfrak{M}^1_k)_\sharp \rho(\mathbf{z}_{k+1}) = \pi\left(\mathbf{z}_{k+1} | \mathbf{y}_{j\in\Xi,\, j\leq k+1}\right)$

Full solution: $(\mathfrak{T}_k)_\sharp \rho(\mathbf{z}_{0:k+1}) = \pi\left(\mathbf{z}_{0:k+1} | \mathbf{y}_{\Xi\leq k+1}\right)$ where,

$$\mathfrak{T}_k := \begin{bmatrix} \mathfrak{M}_0(\mathbf{z}_0, \mathbf{z}_1) \\ \mathbf{z}_2 \\ \mathbf{z}_3 \\ \vdots \\ \mathbf{z}_{k+1} \end{bmatrix} \circ \begin{bmatrix} \mathbf{z}_0 \\ \mathfrak{M}_1(\mathbf{z}_1, \mathbf{z}_2) \\ \mathbf{z}_3 \\ \vdots \\ \mathbf{z}_{k+1} \end{bmatrix} \circ \cdots \circ \begin{bmatrix} \mathbf{z}_0 \\ \vdots \\ \mathbf{z}_{k-2} \\ \mathbf{z}_{k-1} \\ \mathfrak{M}_k(\mathbf{z}_k, \mathbf{z}_{k+1}) \end{bmatrix}$$

Removing conditional dependencies – step 1

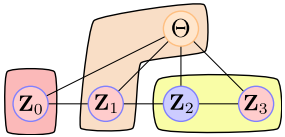
Find the map

$$\mathfrak{M}_0(\boldsymbol{\theta}, \mathbf{z}_0, \mathbf{z}_1) = \begin{bmatrix} \mathfrak{M}_0^{\Theta}(\boldsymbol{\theta}) \\ \mathfrak{M}_0^{0}(\boldsymbol{\theta}, \mathbf{z}_0, \mathbf{z}_1) \\ \mathfrak{M}_0^{1}(\boldsymbol{\theta}, \mathbf{z}_1) \end{bmatrix}$$

pushing forward $\mathcal{N}(0, \mathbf{I})$ to the **first Markov component** of $\pi\left(\Theta, \mathbf{Z}_\Lambda \middle| \mathbf{y}_\Xi\right)$:

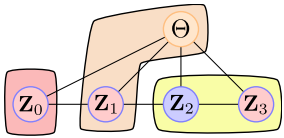
$$\pi^0(\Theta, \mathbf{Z}_0, \mathbf{Z}_1) := \mathcal{L}(\mathbf{y}_0|\Theta, \mathbf{Z}_0)\mathcal{L}(\mathbf{y}_1|\Theta, \mathbf{Z}_1)\pi(\mathbf{Z}_1|\Theta, \mathbf{Z}_0)\pi(\mathbf{Z}_0|\Theta)\pi(\Theta)$$

π :



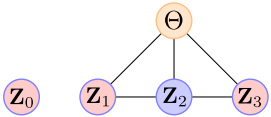
Removing conditional dependencies – step 1

π :



$$T_0(\boldsymbol{\theta}, \mathbf{z}_\Lambda) = \begin{bmatrix} \mathfrak{M}_0(\boldsymbol{\theta}, \mathbf{z}_0, \mathbf{z}_1) \\ \mathbf{z}_2 \\ \mathbf{z}_3 \end{bmatrix}$$

$T_0^\sharp \pi$:



Removing conditional dependencies – step 2

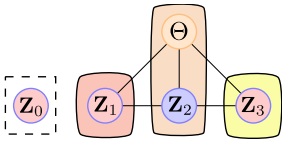
Find the map

$$\mathfrak{M}_1(\boldsymbol{\theta}, \mathbf{z}_1, \mathbf{z}_2) = \begin{bmatrix} \mathfrak{M}_1^{\Theta}(\boldsymbol{\theta}) \\ \mathfrak{M}_1^{0}(\boldsymbol{\theta}, \mathbf{z}_1, \mathbf{z}_2) \\ \mathfrak{M}_1^{1}(\boldsymbol{\theta}, \mathbf{z}_2) \end{bmatrix}$$

pushing $\mathcal{N}(0, \mathbf{I})$ to the **second Markov component** of $T_0^\sharp \pi$:

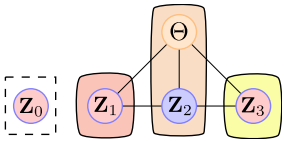
$$\pi^1(\Theta, \mathbf{Z}_1, \mathbf{Z}_2) := \rho(\Theta, \mathbf{Z}_1)\, \pi\left(\mathbf{Z}_2 \,\middle|\, \Theta', \mathbf{Z}_1'\right) \quad \text{(missing observation)}$$
$$\Theta' := \mathfrak{M}_0^{\Theta}(\Theta)\,, \quad \mathbf{Z}_1' := \mathfrak{M}_0^{1}(\Theta, \mathbf{Z}_1)$$

$T_0^\sharp \pi$:



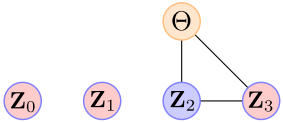
Removing conditional dependencies – step 2

$T_0^\sharp \pi$:



$$T_1(\boldsymbol{\theta}, \mathbf{z}_\Lambda) = \begin{bmatrix} \mathfrak{M}_1^\Theta(\boldsymbol{\theta}) \\ \mathbf{z}_0 \\ \mathfrak{M}_1^0(\boldsymbol{\theta}, \mathbf{z}_1, \mathbf{z}_2) \\ \mathfrak{M}_1^1(\boldsymbol{\theta}, \mathbf{z}_2) \\ \mathbf{z}_3 \end{bmatrix}$$

$T_1^\sharp T_0^\sharp \pi$:



Removing conditional dependencies – step 3

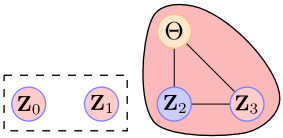
Find the map

$$\mathfrak{M}_2(\boldsymbol{\theta}, \mathbf{z}_2, \mathbf{z}_3) = \begin{bmatrix} \mathfrak{M}_2^{\Theta}(\boldsymbol{\theta}) \\ \mathfrak{M}_2^{0}(\boldsymbol{\theta}, \mathbf{z}_2, \mathbf{z}_3) \\ \mathfrak{M}_2^{1}(\boldsymbol{\theta}, \mathbf{z}_3) \end{bmatrix}$$

pushing $\mathcal{N}(0, \mathbf{I})$ to the **third Markov component** of $T_1^\sharp T_0^\sharp \pi$:

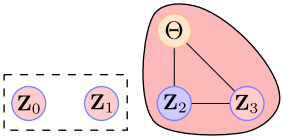
$$\pi^2(\Theta, \mathbf{Z}_2, \mathbf{Z}_3) := \rho(\Theta, \mathbf{Z}_2)\, \mathcal{L}\left(\mathbf{y}_3 \,\middle|\, \Theta'', \mathbf{Z}_3\right) \pi\left(\mathbf{Z}_3 \,\middle|\, \Theta'', \mathbf{Z}_2''\right)$$
$$\Theta'' := \mathfrak{M}_0^{\Theta} \circ \mathfrak{M}_1^{\Theta}(\Theta)\,, \quad \mathbf{Z}_1'' := \mathfrak{M}_1^{1}(\Theta, \mathbf{Z}_2)$$

$T_1^\sharp T_0^\sharp \pi$:



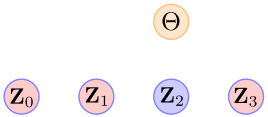
Removing conditional dependencies – step 3

$T_1^\sharp T_0^\sharp \pi$:



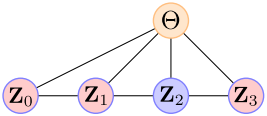
$$T_2(\boldsymbol{\theta}, \mathbf{z}_\Lambda) = \begin{bmatrix} \mathfrak{M}_2^\Theta(\boldsymbol{\theta}) \\ \mathbf{z}_{0:1} \\ \mathfrak{M}_2^0(\boldsymbol{\theta}, \mathbf{z}_2, \mathbf{z}_3) \\ \mathfrak{M}_2^1(\boldsymbol{\theta}, \mathbf{z}_3) \end{bmatrix}$$

$T_2^\sharp T_1^\sharp T_0^\sharp \pi$:



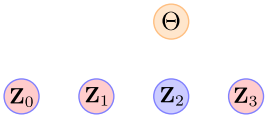
Removing conditional dependencies

π :



$$\mathfrak{T} = \begin{bmatrix} \mathfrak{M}_0^{\Theta}(\boldsymbol{\theta}) \\ \mathfrak{M}_0^0(\boldsymbol{\theta}, \mathbf{z}_0, \mathbf{z}_1) \\ \mathfrak{M}_0^1(\boldsymbol{\theta}, \mathbf{z}_1) \\ \mathbf{z}_2 \\ \mathbf{z}_3 \end{bmatrix} \circ \begin{bmatrix} \mathfrak{M}_1^{\Theta}(\boldsymbol{\theta}) \\ \mathbf{z}_0 \\ \mathfrak{M}_1^0(\boldsymbol{\theta}, \mathbf{z}_1, \mathbf{z}_2) \\ \mathfrak{M}_1^1(\boldsymbol{\theta}, \mathbf{z}_2) \\ \mathbf{z}_3 \end{bmatrix} \circ \begin{bmatrix} \mathfrak{M}_1^{\Theta}(\boldsymbol{\theta}) \\ \mathbf{z}_0 \\ \mathbf{z}_1 \\ \mathfrak{M}_2^0(\boldsymbol{\theta}, \mathbf{z}_2, \mathbf{z}_3) \\ \mathfrak{M}_2^1(\boldsymbol{\theta}, \mathbf{z}_3) \end{bmatrix}$$

$\mathfrak{T}^\sharp\pi$:



A framework for sequential inference

- **Variational Bayesian algorithms** for joint state-parameter inference
- Yields **filtering**, **smoothing**, **full posterior** via composition of maps
- Constant cost per assimilation/prediction step
- Evaluation of smoothing/full posterior grows **linearly** with time
- Map parametrization *(basis, sparsity)* governs **complexity/accuracy trade-off**

A framework for sequential inference

- **Variational Bayesian algorithms** for joint state-parameter inference
- Yields **filtering**, **smoothing**, **full posterior** via composition of maps
- Constant cost per assimilation/prediction step
- Evaluation of smoothing/full posterior grows **linearly** with time
- Map parametrization *(basis, sparsity)* governs **complexity/accuracy trade-off**

We need to characterize maps between $N(0, \mathbf{I})$
and the lag-1 smoothing

Preconditioning the assimilation step

At step k one needs to solve the problem

$$\mathfrak{M}_k = \operatorname*{arg\,min}_{\mathfrak{M}\in\mathcal{T}_>} \mathcal{D}_{\mathsf{KL}}\left(\mathfrak{M}_\sharp\rho(\mathbf{z}_k,\mathbf{z}_{k+1})\middle\|\underbrace{\begin{bmatrix}\mathfrak{M}^1_{k-1}(\mathbf{z}_k)\\ \mathbf{z}_{k+1}\end{bmatrix}^\sharp \pi(\mathbf{z}_k,\mathbf{z}_{k+1}|\mathbf{y}_{j\le k+1})}_{\mathcal{L}(\mathbf{y}_{k+1}|\mathbf{z_{k+1}})\,\pi\left(\mathbf{z}_{k+1}\middle|\mathfrak{M}^1_{k-1}(\mathbf{z}_k)\right)\rho(\mathbf{z}_k)}\right).$$

Preconditioning the assimilation step

At step k one needs to solve the problem

$$\mathfrak{M}_k = \operatorname*{arg\,min}_{\mathfrak{M}\in\mathcal{T}_>} \mathcal{D}_{\mathsf{KL}}\left(\mathfrak{M}_\sharp\rho(\mathbf{z}_k,\mathbf{z}_{k+1})\middle\|\underbrace{\begin{bmatrix}\mathfrak{M}^1_{k-1}(\mathbf{z}_k)\\ \mathbf{z}_{k+1}\end{bmatrix}^\sharp \pi(\mathbf{z}_k,\mathbf{z}_{k+1}|\mathbf{y}_{j\le k+1})}_{\mathcal{L}(\mathbf{y}_{k+1}|\mathbf{z_{k+1}})\,\pi\left(\mathbf{z}_{k+1}\middle|\mathfrak{M}^1_{k-1}(\mathbf{z}_k)\right)\rho(\mathbf{z}_k)}\right).$$

Ansatz: the next filtering distribution is similar to the latest.

Preconditioning the assimilation step

At step k one needs to solve the problem

$$\mathfrak{M}_k = \operatorname*{arg\,min}_{\mathfrak{M}\in\mathcal{T}_>} \mathcal{D}_{\mathsf{KL}}\left(\mathfrak{M}_\sharp\rho(\mathbf{z}_k,\mathbf{z}_{k+1})\middle\|\underbrace{\begin{bmatrix}\mathfrak{M}^1_{k-1}(\mathbf{z}_k)\\ \mathbf{z}_{k+1}\end{bmatrix}^\sharp \pi(\mathbf{z}_k,\mathbf{z}_{k+1}|\mathbf{y}_{j\le k+1})}_{\mathcal{L}(\mathbf{y}_{k+1}|\mathbf{z_{k+1}})\,\pi\left(\mathbf{z}_{k+1}\middle|\mathfrak{M}^1_{k-1}(\mathbf{z}_k)\right)\rho(\mathbf{z}_k)}\right).$$

Ansatz: the next filtering distribution is similar to the latest.

$$\widetilde{\mathfrak{M}}_k = \operatorname*{arg\,min}_{\mathfrak{M}\in\mathcal{T}_>} \mathcal{D}_{\mathsf{KL}}\left(\mathfrak{M}_\sharp\rho(\mathbf{z}_k,\mathbf{z}_{k+1})\middle\|\begin{bmatrix}\mathfrak{M}^1_{k-1}(\mathbf{z}_k)\\ \mathfrak{M}^1_{k-1}(\mathbf{z}_{k+1})\end{bmatrix}^\sharp \pi(\mathbf{z}_k,\mathbf{z}_{k+1}|\mathbf{y}_{j\le k+1})\right).$$

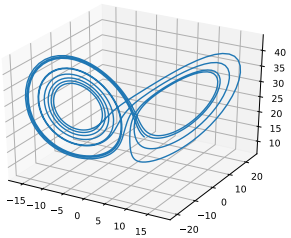
$$\mathfrak{M}^1_k(\mathbf{z}_{k+1}) = \operatorname*{arg\,min}_{\mathfrak{M}^1\in\mathcal{T}_>}\left\|\mathfrak{M}^1(\mathbf{z}_{k+1}) - \mathfrak{M}^1_{k-1}\circ\widetilde{\mathfrak{M}}^1_k(\mathbf{z}_{k+1})\right\|$$

Lorenz 63

Dynamics

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

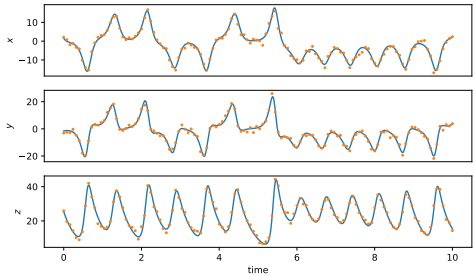
$$\sigma = 10,\ \beta = 8/3,\ \rho = 28$$



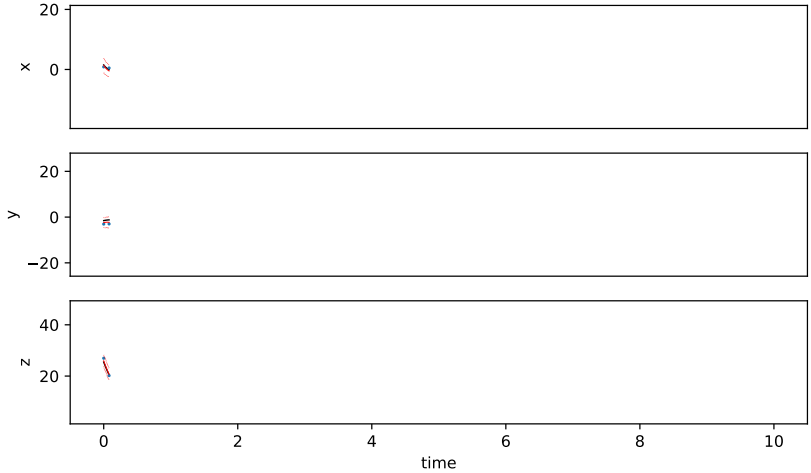
Assimilation model

$$\mathbf{u}_{k+1} = \mathbf{u}_k + 0.01 \cdot \mathcal{F}(\mathbf{u}_k) + \mathcal{N}(0, 10^{-1} \cdot \mathbf{I})$$

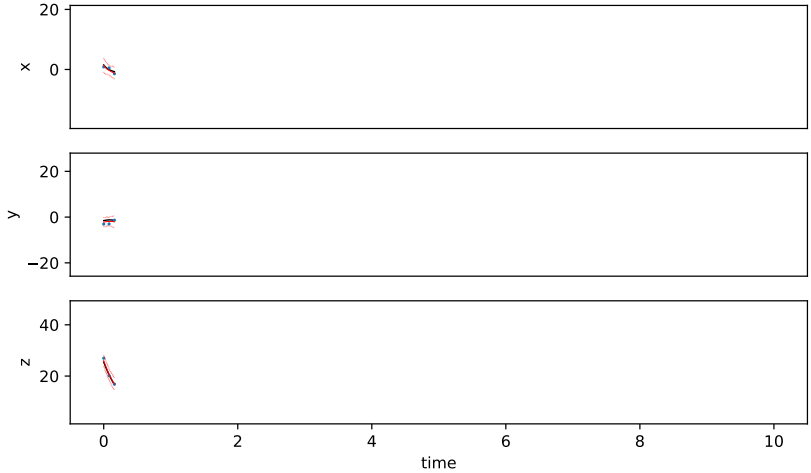
$$\mathbf{h}_k = \mathcal{O}(\mathbf{u}_k) + \mathcal{N}(0, 2 \cdot \mathbf{I}) \ , \quad k = 0, 8, 16, \ldots$$



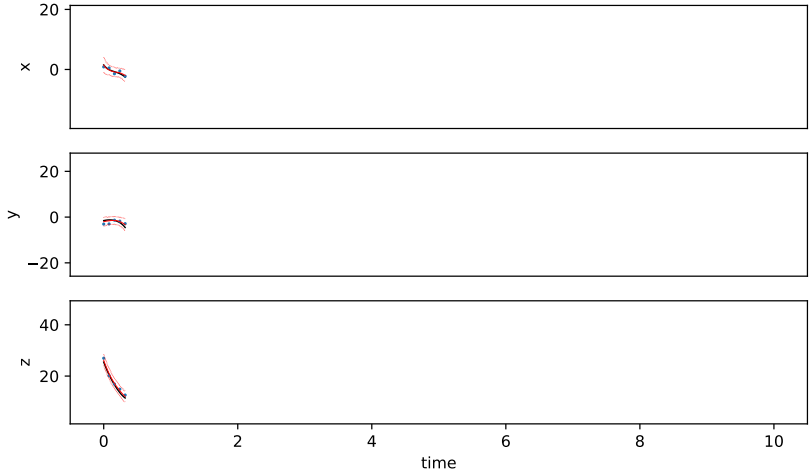
Smoothing



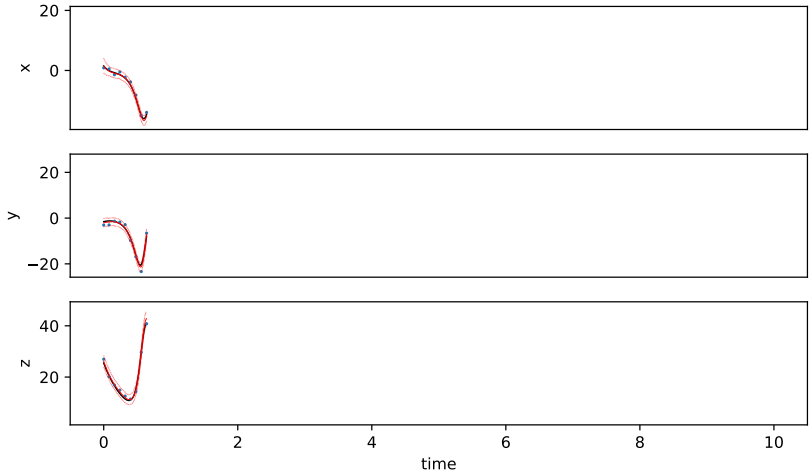
Smoothing



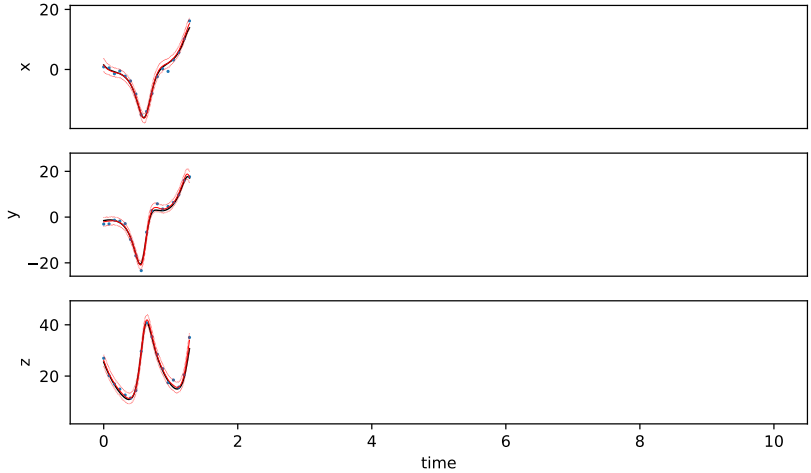
Smoothing



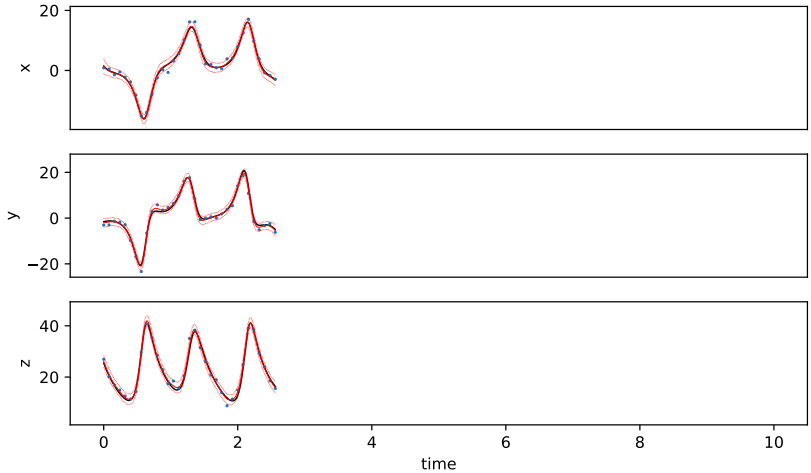
Smoothing



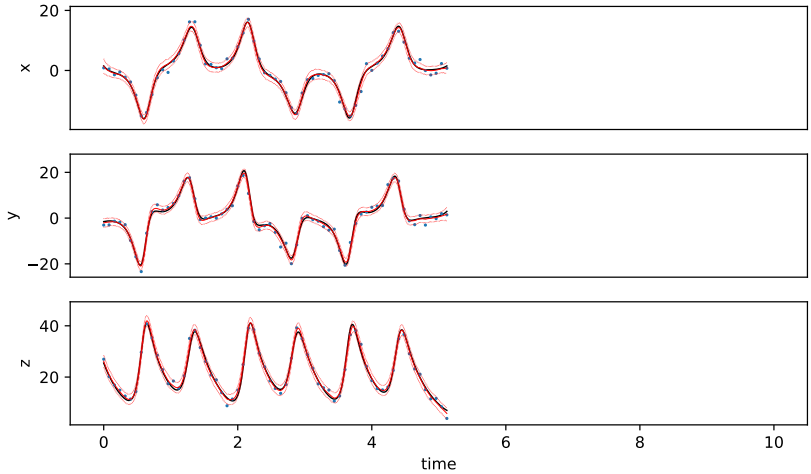
Smoothing



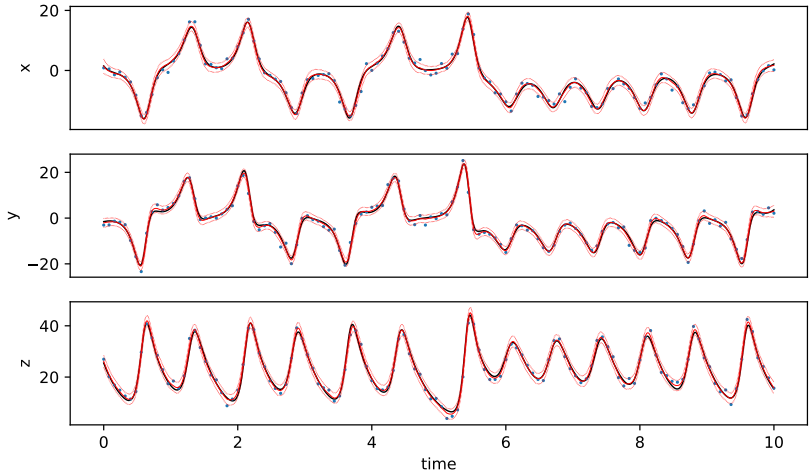
Smoothing



Smoothing

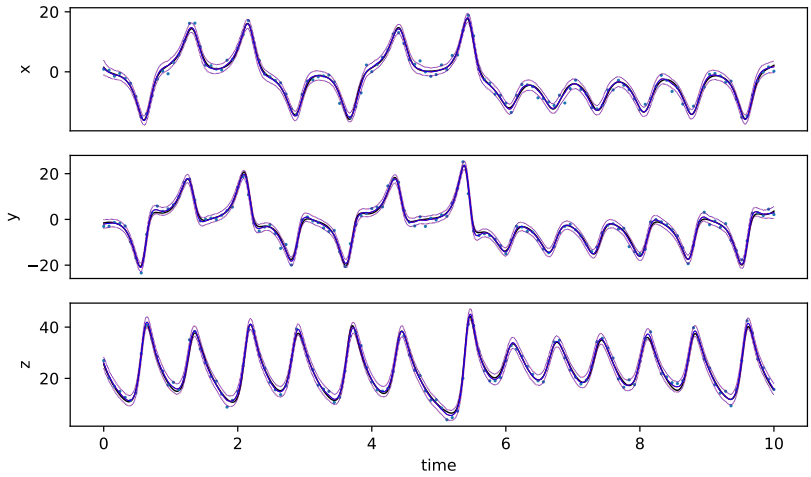


Smoothing



		filtering		smoothing	
		Order 1	Order 3	Order 1	Order 3
at obs.	Avg. RMSE	7.39×10^{-1}	7.46×10^{-1}	4.82×10^{-1}	4.86×10^{-1}
	Med. RMSE	6.49×10^{-1}	6.61×10^{-1}	4.18×10^{-1}	4.23×10^{-1}
	Var. RMSE	1.51×10^{-1}	1.55×10^{-1}	5.62×10^{-2}	5.78×10^{-2}
incl. pred.	Avg. RMSE	8.37×10^{-1}	8.42×10^{-1}	4.67×10^{-1}	4.72×10^{-1}
	Med. RMSE	7.54×10^{-1}	7.50×10^{-1}	4.11×10^{-1}	4.14×10^{-1}
	Var. RMSE	2.23×10^{-1}	2.25×10^{-1}	5.53×10^{-2}	5.72×10^{-2}

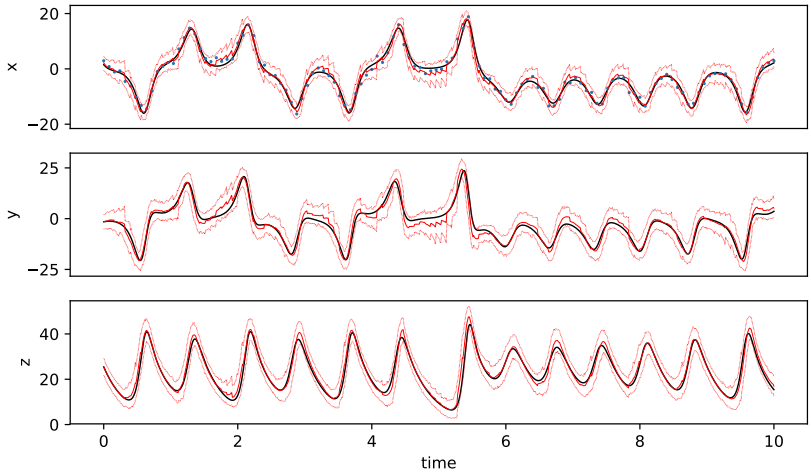
Smoothing – maps vs. MCMC



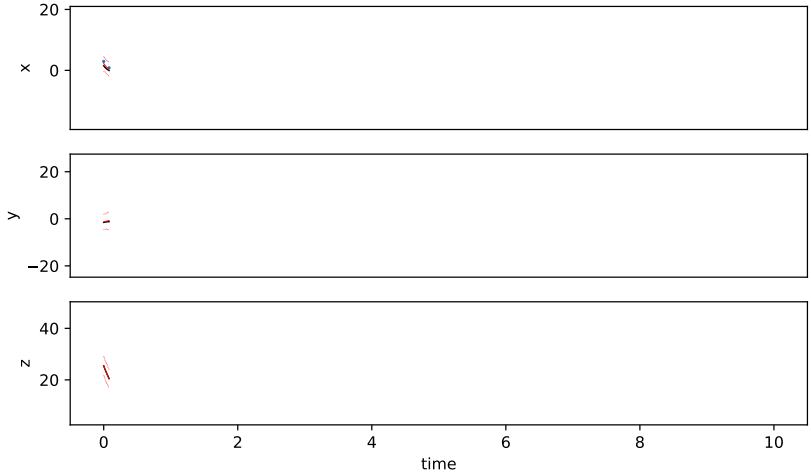
		filtering		smoothing	
		Order 1	Order 3	Order 1	Order 3
at obs.	Avg. RMSE	7.39×10^{-1}	7.46×10^{-1}	4.82×10^{-1}	4.86×10^{-1}
	Med. RMSE	6.49×10^{-1}	6.61×10^{-1}	4.18×10^{-1}	4.23×10^{-1}
	Var. RMSE	1.51×10^{-1}	1.55×10^{-1}	5.62×10^{-2}	5.78×10^{-2}

Posterior accuracy ($d = 3000$)	Order 1	Order 3
$\mathbb{V}[\log \frac{\rho}{T^\sharp \pi}]$	8.58×10^{-1}	2.11×10^{-1}
Metropolis indep. A/R	33.7%	62.9%
Metropolis indep. ESS (worst)	0.68%	12.54%

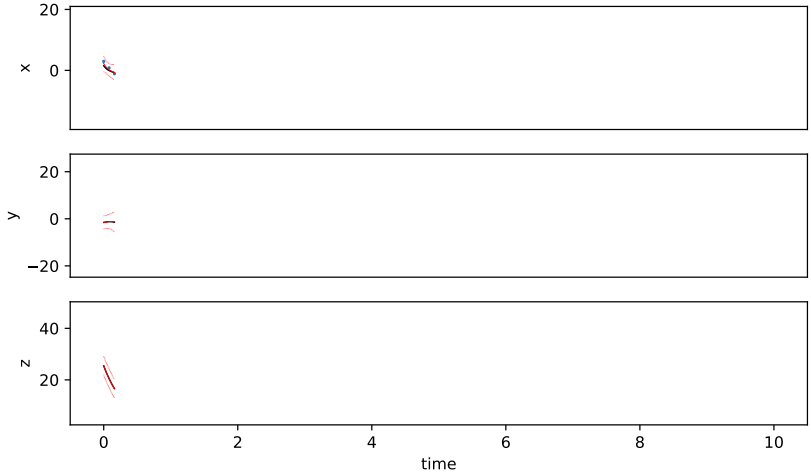
Filtering



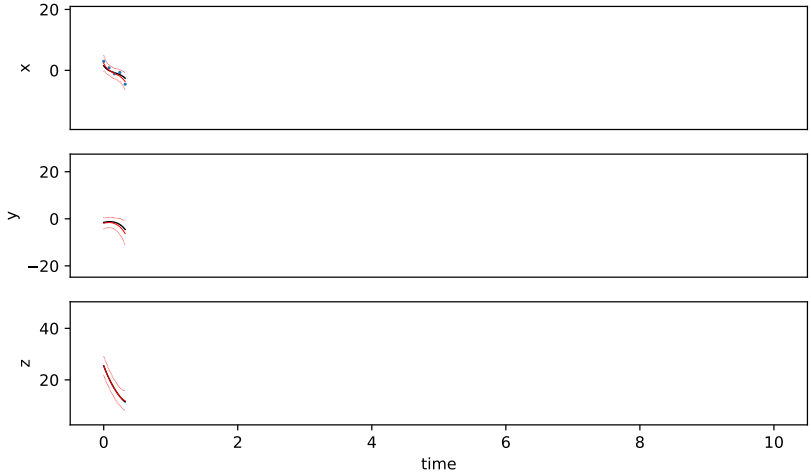
Smoothing



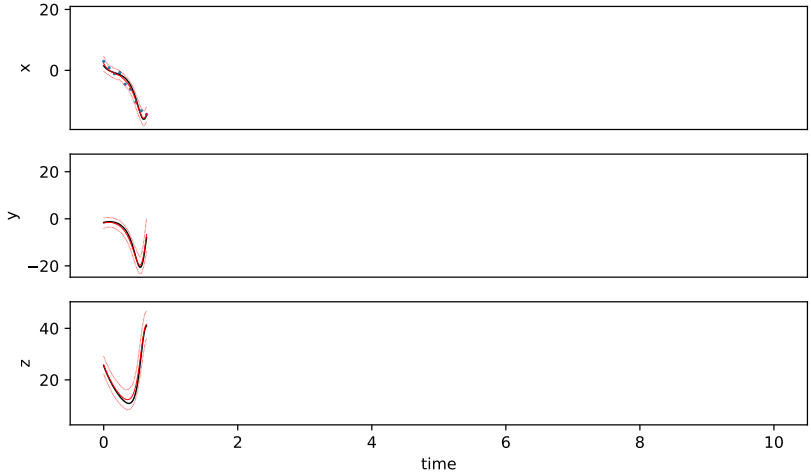
Smoothing



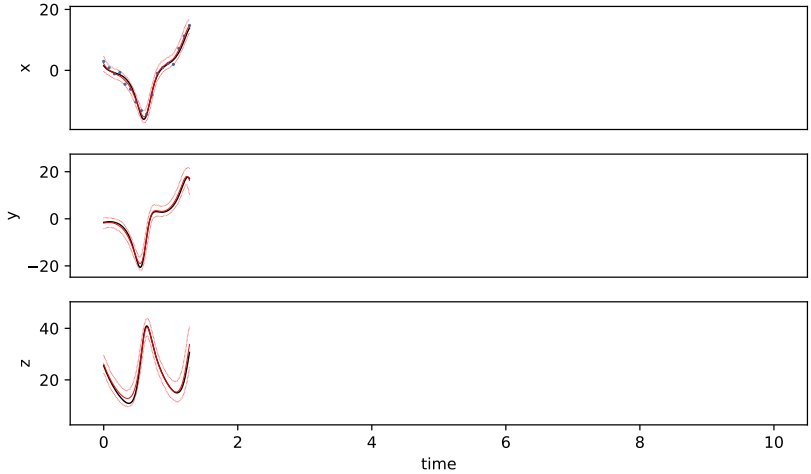
Smoothing



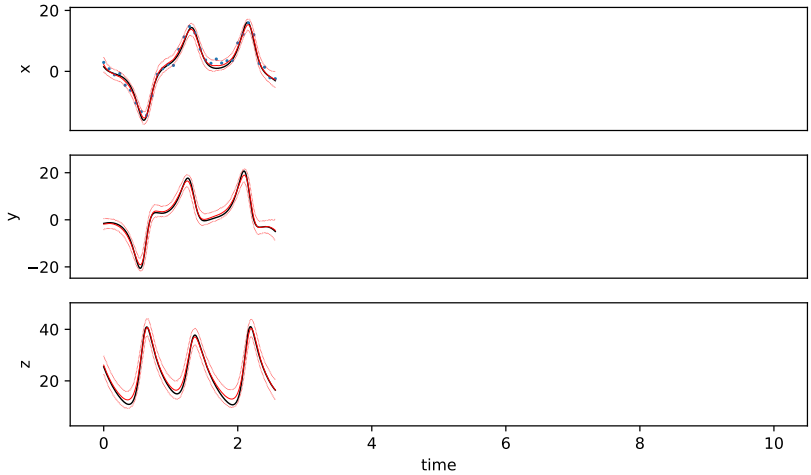
Smoothing



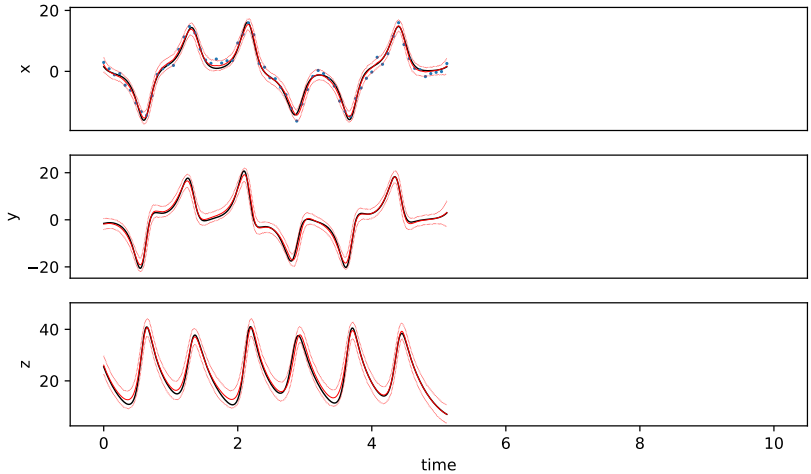
Smoothing



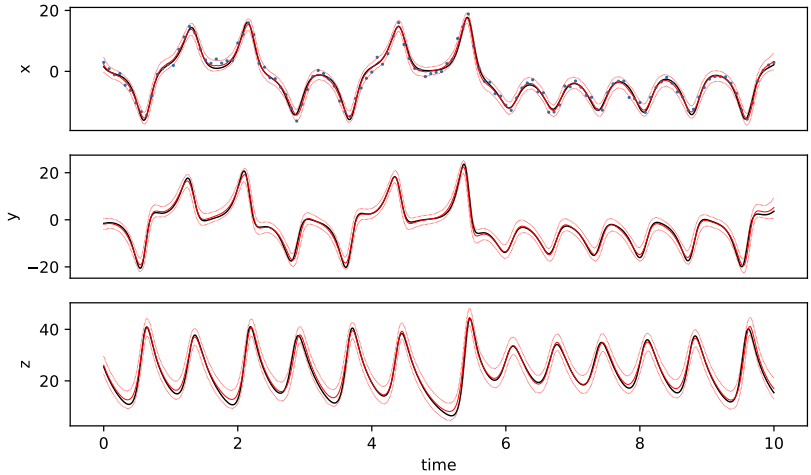
Smoothing



Smoothing



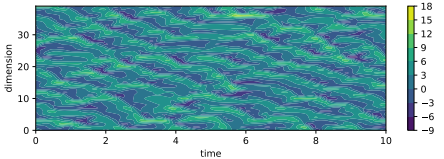
Smoothing



Lorenz 96

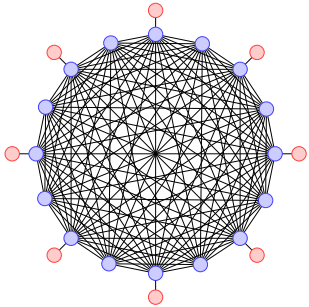
Dynamics

$$\frac{d\mathbf{Z}_j}{dt} = (\mathbf{Z}_{j+1} - \mathbf{Z}_{j-2})\,\mathbf{Z}_{j-1} - \mathbf{Z}_j + F$$
$$j \in \{1, \dots, 40\}\,, \quad F = 8 \text{ (chaotic)}$$

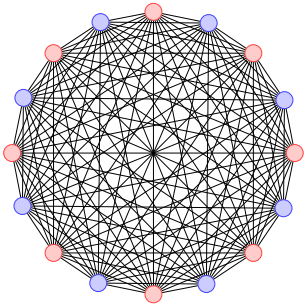


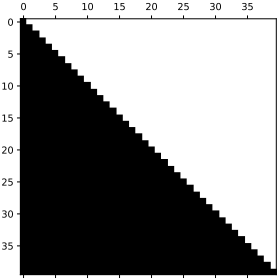
Assimilation model

$$\mathbf{u}_{k+1} = \mathbf{u}_k + 0.01 \cdot \mathcal{F}(\mathbf{u}_k) + \mathcal{N}(0, 10^{-1} \cdot \mathbf{I})$$
$$\mathbf{h}_k = \mathcal{O}(\mathbf{u}_k) + \mathcal{N}(0, 0.5 \cdot \mathbf{I})\,,$$
$$k = 0, 10, \dots$$

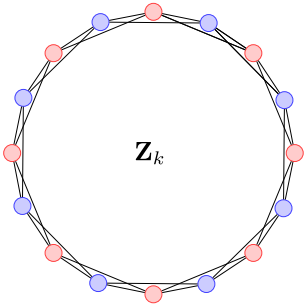


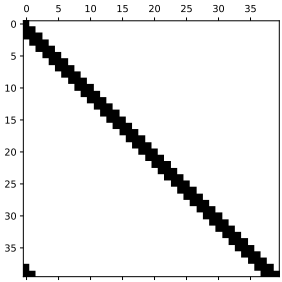
Map sparsity ansatz



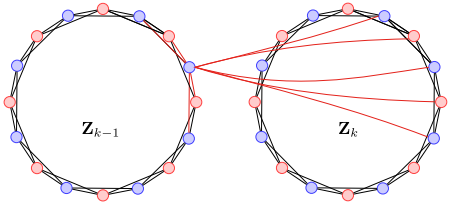


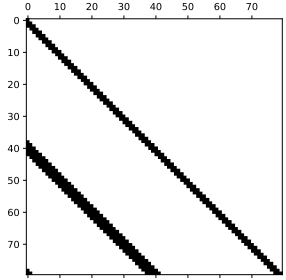
Map sparsity ansatz



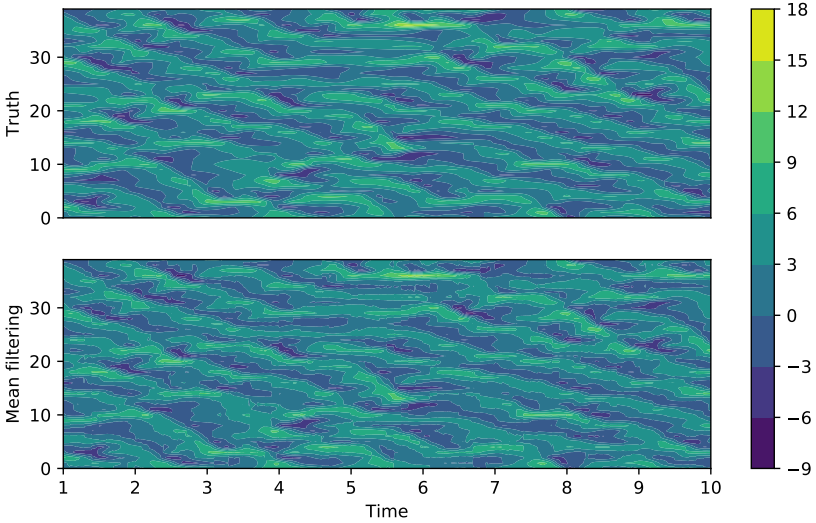


Map sparsity ansatz

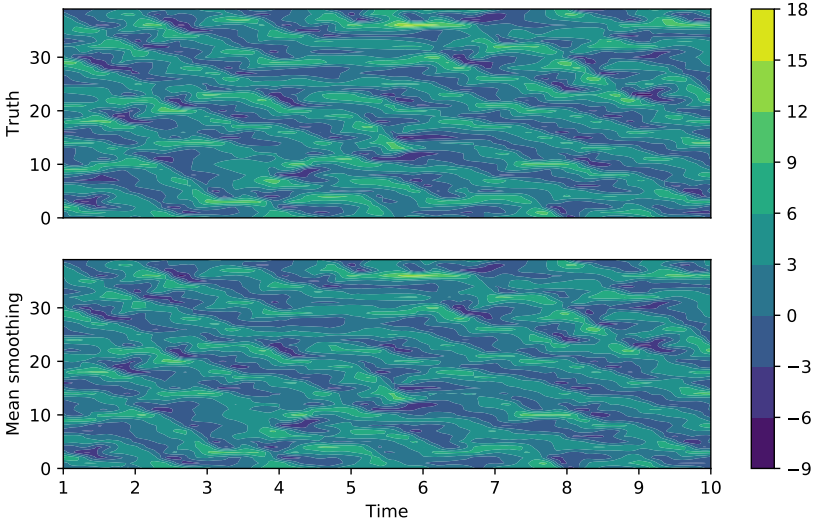




Filtering


Truth
Mean filtering
0
10
20
30
1
2
3
4
5
6
7
8
9
10
Time
18
15
12
9
6
3
0
−3
−6
−9

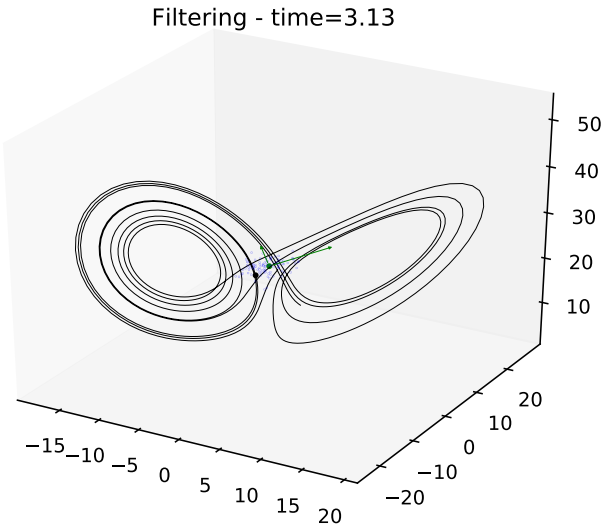
Smoothing


Truth
Mean smoothing
0
10
20
30
1
2
3
4
5
6
7
8
9
10
Time
18
15
12
9
6
3
0
−3
−6
−9

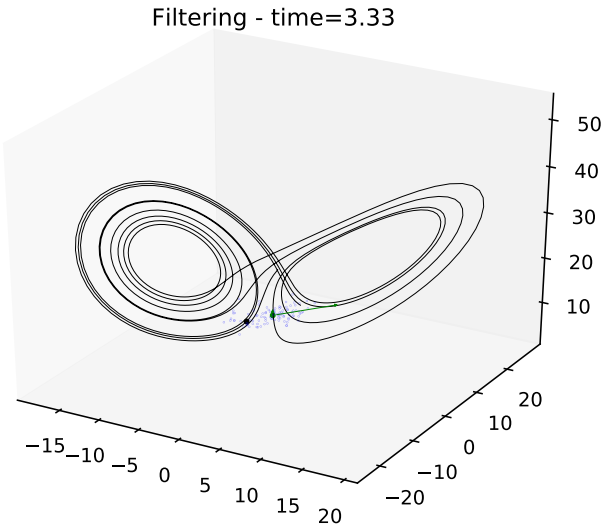
		Order 1	
		Filtering	Smoothing ($d = 4 \times 10^4$)
at obs.	Avg. RMSE	8.35×10^{-1}	7.40×10^{-1}
	Med. RMSE	8.11×10^{-1}	7.29×10^{-1}
	Var. RMSE	3.26×10^{-2}	2.42×10^{-2}
incl. pred.	Avg. RMSE	9.20×10^{-1}	7.34×10^{-1}
	Med. RMSE	9.01×10^{-1}	7.25×10^{-1}
	Var. RMSE	4.21×10^{-2}	2.51×10^{-2}

Towards high(er)-dimensions

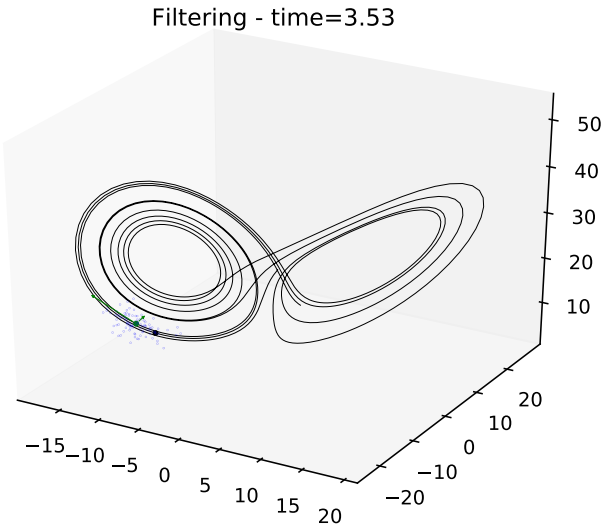
Updates act only on low-dimensional subsapces



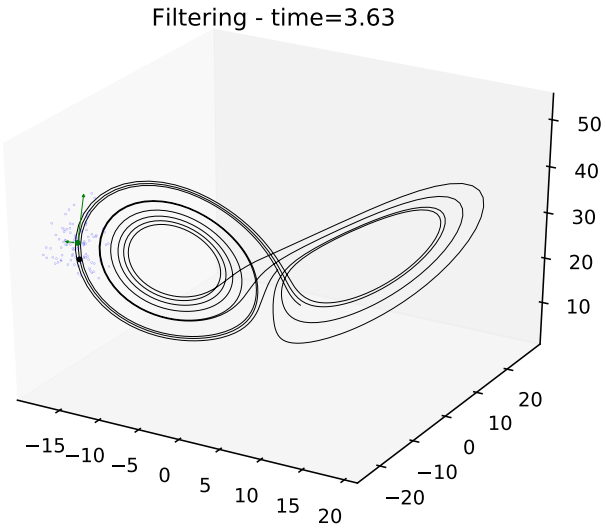
Updates act only on low-dimensional subsapces



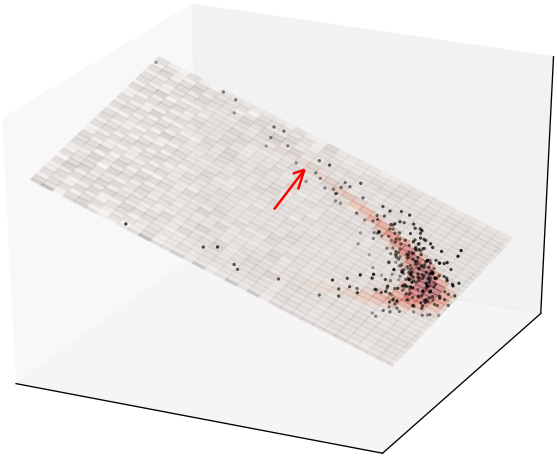
Updates act only on low-dimensional subsapces



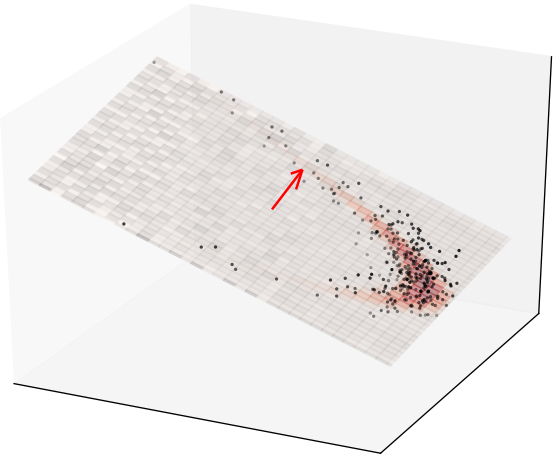
Updates act only on low-dimensional subsapces



Work only on directions that depart from the reference



Work only on directions that depart from the reference



How to detect such directions?

Work with lazy maps [B, Zahm, Spantini, Marzouk 2019]

$$\widetilde{\mathfrak{M}}_k = \operatorname*{arg\,min}_{\mathfrak{M} \in \mathcal{T}_>} \mathcal{D}_{\mathsf{KL}} \left(\mathfrak{M}_\sharp \rho(\mathbf{z}_k, \mathbf{z}_{k+1}) \middle\| \underbrace{\begin{bmatrix} \mathfrak{M}_{k-1}^1(\mathbf{z}_k) \\ \mathfrak{M}_{k-1}^1(\mathbf{z}_{k+1}) \end{bmatrix}^\sharp \pi(\mathbf{z}_k, \mathbf{z}_{k+1} | \mathbf{y}_{j \le k+1})}_{\widetilde{\pi}(\mathbf{z}_k, \mathbf{z}_{k+1})} \right)$$

For $\mathfrak{r}(\boldsymbol{x}) = \log(\widetilde{\pi}(\boldsymbol{x})/\rho(\boldsymbol{x}))$ let

$$\mathbf{H} = \int \nabla \mathfrak{r}(\boldsymbol{x}) \, \nabla \mathfrak{r}(\boldsymbol{x})^\top \, \rho(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \, , \quad \text{and} \quad \mathbf{H}\mathbf{u}_i = \lambda_i \mathbf{u}_i \, .$$

Then, for $\lambda_i \geq \lambda_{i+1}$, $\mathbf{U}_r = [\mathbf{u}_1, \dots, \mathbf{u}_{i+1}]$ and $\mathbf{Q} = \mathbf{U}_r | \mathbf{U}_\perp$, there exist

$$\mathfrak{M}(\mathbf{z}) = \begin{bmatrix} \mathfrak{M}_r(\mathbf{z}_{1:r}) \\ \mathbf{z}_{r+1:d} \end{bmatrix} \in \mathcal{T}_>^r \quad \text{such that} \quad \mathfrak{M}_\sharp \rho = \mathbf{Q}^\sharp \widetilde{\pi} \, .$$

Work with lazy maps [B, Zahm, Spantini, Marzouk 2019]

$$\widetilde{\mathfrak{M}}_k = \underset{\mathfrak{M}\in\mathcal{T}_>}{\arg\min}\, \mathcal{D}_{\mathsf{KL}}\left(\mathfrak{M}_\sharp\rho(\mathbf{z}_k,\mathbf{z}_{k+1})\middle\| \underbrace{\begin{bmatrix}\mathfrak{M}_{k-1}^1(\mathbf{z}_k)\\ \mathfrak{M}_{k-1}^1(\mathbf{z}_{k+1})\end{bmatrix}^\sharp \pi(\mathbf{z}_k,\mathbf{z}_{k+1}|\mathbf{y}_{j\le k+1})}_{\widetilde{\pi}(\mathbf{z}_k,\mathbf{z}_{k+1})}\right)$$

For $\mathfrak{r}(\boldsymbol{x}) = \log(\widetilde{\pi}(\boldsymbol{x})/\rho(\boldsymbol{x}))$ let

$$\mathbf{H} = \int \nabla\mathfrak{r}(\boldsymbol{x})\ \nabla\mathfrak{r}(\boldsymbol{x})^\top\ \rho(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x}\ , \quad \text{and} \quad \mathbf{H}\mathbf{u}_i = \lambda_i\mathbf{u}_i\ .$$

Then, for $\lambda_i \ge \lambda_{i+1}$, $\mathbf{U}_r = [\mathbf{u}_1,\dots,\mathbf{u}_{i+1}]$ and $\mathbf{Q} = \mathbf{U}_r|\mathbf{U}_\perp$, there exist

$$\mathfrak{M}(\mathbf{z}) = \begin{bmatrix}\mathfrak{M}_r(\mathbf{z}_{1:r})\\ \mathbf{z}_{r+1:d}\end{bmatrix} \in \mathcal{T}_>^r \quad \text{such that} \quad \mathfrak{M}_\sharp\rho = \mathbf{Q}^\sharp\widetilde{\pi}\ .$$

This would be perfect is $\mathbf{Q}$ was triangular.
But it is NOT!

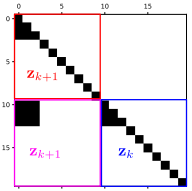
Build state-dependent subspaces

1. Assemble

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{k,k} & \mathbf{H}_{k,k+1} \\ \mathbf{H}_{k+1,k} & \mathbf{H}_{k+1,k+1} \end{bmatrix}, \qquad \mathbf{H}_{i,j} = \int \left(\nabla_{\mathbf{z}_i} \mathfrak{r}\right)\left(\nabla_{\mathbf{z}_j} \mathfrak{r}\right)^\top \rho(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}$$

2. Solve two truncated singular value problems:
   - $[\mathbf{H}_{k,k} | \mathbf{H}_{k,k+1}] = \mathbf{U}_{r_k} \Sigma_{1:r_k} \mathbf{V}_{r_k}$
   - $[\mathbf{H}_{k+1,k} | \mathbf{H}_{k+1,k+1}] = \mathbf{U}_{r_{k+1}} \Sigma_{1:r_{k+1}} \mathbf{V}_{r_{k+1}}$
3. Find a basis $\mathbf{U}_r$ for $\mathrm{span}(\mathbf{U}_{r_k} | \mathbf{U}_{r_{k+1}})$, e.g. using the QR decomposition
4. Set $\mathbf{Q} = \begin{bmatrix} \mathbf{U}_r | \mathbf{U}_\perp & 0 \\ 0 & \mathbf{U}_r | \mathbf{U}_\perp \end{bmatrix}$
5. Solve $\widetilde{\mathfrak{M}}_k = \arg\min_{\mathfrak{M} \in \mathcal{T}^r_>} \mathcal{D}_{\mathsf{KL}} \left(\mathfrak{M}_\sharp \rho(\mathbf{z}_k, \mathbf{z}_{k+1}) \middle\| \mathbf{Q}^\sharp \widetilde{\pi}(\mathbf{z}_k, \mathbf{z}_{k+1}) \right)$

Sparsity of $\widetilde{\mathfrak{M}}_k$:



Lorenz 96 – model II

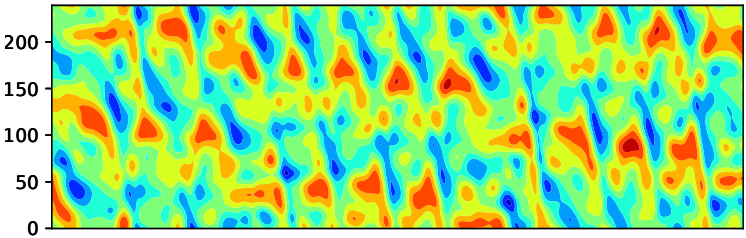
Dynamics

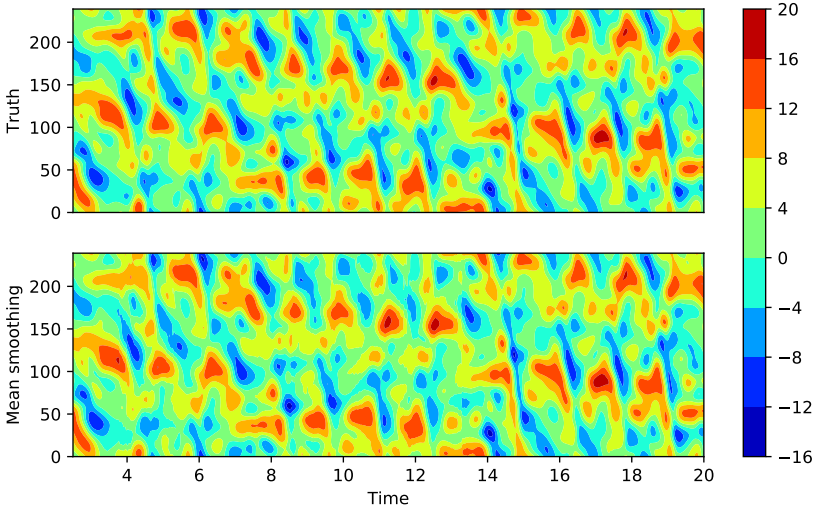
$$\frac{d\mathbf{X}_n}{dt} = \sum_{j=-J}^{J} \sum_{i=-J}^{J} \left(X_{n-K+j-i} X_{n+K+j} - X_{n-2K-i} X_{n-K-j}\right) / K^2 - X_n + F$$

$\mathbf{X} \in \mathbb{R}^d$, $d = 240$, $K = 33$, $J = (K-1)/2$, $F = 14$ (chaotic)

Assimilation model

$$\mathbf{u}_{k+1} = \mathbf{u}_k + 0.025 \cdot \mathcal{F}(\mathbf{u}_k) + \mathcal{N}(0, 10^{-1} \cdot \mathbf{I})$$
$$\mathbf{h}_i = \mathcal{O}(\mathbf{u}_i) + \mathcal{N}(0, 10^{-1} \cdot \mathbf{I}) \ , \ i = 0, 4, \ldots$$





Using rank-2 lazy maps of order 2 and 50 samples.

Key contributions

Algorithms for filtering, smoothing, and parameters estimation via **deterministic couplings** and **optimization**.

Contact: Daniele Bigoni – **dabi@mit.edu**

Software: `https://transportmaps.mit.edu`

Bigoni et al. "Greedy inference with layers of lazy maps" (arXiv)
Bigoni et al. "Adaptive construction of measure transports for Bayesian inference"
Spantini et al. "Inference via low-dimensional couplings" (JMLR)
Marzouk et al. "Sampling via measure transport: an introduction" (Springer)
Parno et al. "Transport map accelerated Markov chain Monte Carlo" (JUQ)
El Moselhy et al. "Bayesian inference with optimal maps" (JCP)

Thanks to: U.S. DEPARTMENT OF ENERGY