Sequential inference via low-dimensional couplings

Alessio Spantini Daniele Bigoni Youssef Marzouk

Massachusetts Institute of Technology
Department of Aeronautics & Astronautics

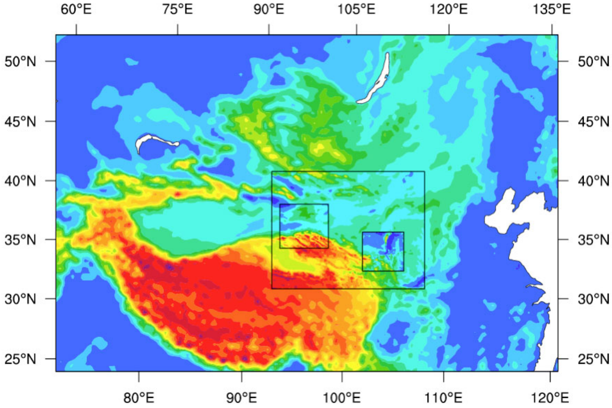
Institute for Mathematics
University of Potsdam

December 1st, 2017

U.S. DEPARTMENT OF ENERGY

Office of Science

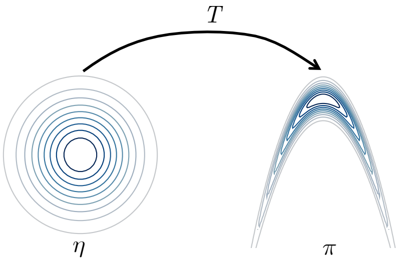
Sequential Bayesian inference



- State estimation (e.g., *filtering* and *smoothing*) or *joint state and parameter estimation*, in a Bayesian setting
  - Need **recursive** algorithms for characterizing the posterior

Deterministic couplings of distributions

Key task: sample a non-Gaussian distribution π

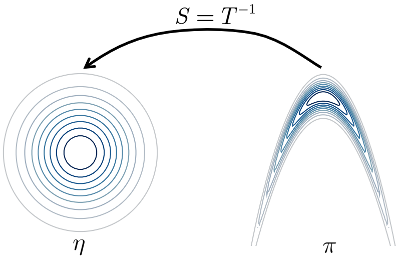


Idea

- Choose a *reference distribution* η (e.g., standard Gaussian)
- Seek a map $T : \mathbb{R}^n \to \mathbb{R}^n$ such that $T_\sharp \eta = \pi$
- Equivalently, find $S = T^{-1}$ such that $S_\sharp \pi = \eta$

Deterministic couplings of distributions

Key task: sample a non-Gaussian distribution π



Idea

- Choose a *reference distribution* η (e.g., standard Gaussian)
- Seek a map $T : \mathbb{R}^n \to \mathbb{R}^n$ such that $T_\sharp \eta = \pi$
- Equivalently, find $S = T^{-1}$ such that $S_\sharp \pi = \eta$

Various types of transport...

- **Optimal transport:**

$$\begin{aligned} T_{\text{opt}} &= \arg\min_{T} \int_{\mathbb{R}^n} c\left(\boldsymbol{x}, T(\boldsymbol{x})\right) \, d\eta(\boldsymbol{x}) \\ & \quad \text{s.t. } T_{\sharp}\,\eta = \pi \end{aligned}$$

  - Monge (1781) problem; many nice properties, but numerically challenging in general continuous cases

- **Knothe-Rosenblatt rearrangement:**

$$T(\boldsymbol{x}) = \begin{bmatrix} T^1(x_1) \\ T^2(x_1, x_2) \\ \vdots \\ T^n(x_1, x_2, \ldots, x_n) \end{bmatrix}$$

  - Exists and is unique (up to ordering) under mild conditions
  - Inverse map $S = T^{-1}$ also lower triangular
  - "Exposes" marginals, will enable conditional sampling...

Computation of triangular maps from densities

$$\min_T \mathcal{D}_{\text{KL}}(\, T_\sharp \eta \,||\, \pi \,)$$

- Numerical approximations can employ a *monotone parameterization*, guaranteeing $\partial_{x_k} T^k > 0$ for arbitrary functions a_k, b_k

$$T^k(x_1, \ldots, x_k) = a_k(x_1, \ldots, x_{k-1}) + \int_0^{x_k} \exp\left(b_k(x_1, \ldots, x_{k-1}, w)\right) dw$$

- Sample average approximation + (*BFGS* or Newton) for

$$\min_{(a_k, b_k)_k} \mathbb{E}_{\boldsymbol{X}\sim\eta}[\, -\log \pi(T(\boldsymbol{X})) \,-\, \sum_k \log \partial_{x_k} T^k(\boldsymbol{X}) \,]$$

- Many alternatives, e.g.,
  1. fully nonparametric approaches (stein variational gradient) [Liu, '16]
  2. deep neural networks (normalizing flows) [Rezende, '15]
- **Challenge:** represent a high-dimensional nonlinear function

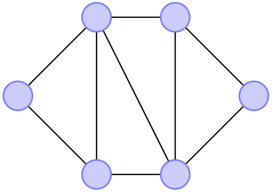
Markov properties and low-dimensional couplings

Main idea

There exists a link between the Markov properties of (η, π) and the existence of couplings that admit low-dimensional structure in terms of

1. Sparsity
2. Decomposability

- Additional structure not discussed here: **low rank**



$$(i,j) \notin \mathcal{E} \quad \text{iff} \quad Z_i \perp\!\!\!\perp Z_j \,|\, \mathbf{Z}_{\mathcal{V}\setminus\{i,j\}}$$

Sparse transport maps

- Given a reference η and a target π, focus on the sparsity of the *inverse* Knothe-Rosenblatt (KR) rearrangement, i.e., $S_{\#}\pi = \eta$

$$S(\boldsymbol{x}) = \begin{bmatrix} S^1(x_1) \\ S^2(x_1, x_2) \\ S^3(x_1, x_2, x_3) \\ \vdots \\ S^n(x_1, x_2, \dots, x_n) \end{bmatrix} \Longrightarrow \begin{bmatrix} S^1(x_1) \\ S^2(x_1, x_2) \\ S^3(\quad x_2, x_3) \\ \vdots \\ S^n(\qquad\qquad x_{n-1}, x_n) \end{bmatrix}$$

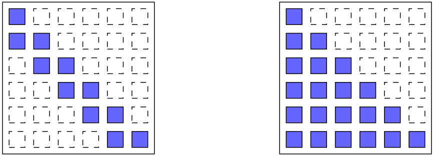
- **Theorem:**[1] The KR rearrangement (a **nonlinear** function) inherits the same sparsity pattern as the Cholesky factor of the incidence matrix (properly scaled) of a graphical model for π, provided that

$$\eta(\mathbf{x}) = \prod_i \eta(x_i)$$

[1]Spantini et al. (2017)

Compute the inverse transport!

- Direct transports $T_{\#}\eta = \pi$, however, tend to be dense
- Sparsity in T is linked to **marginal** (not conditional) independence



Inverse **Direct**

Key message

Compute the inverse transport S and evaluate $T(\mathbf{x}) = S^{-1}(\mathbf{x})$ point-wise

- Trivial to invert a triangular function (sequence of 1D root findings)
- Same spirit as GMRF, but for general **non-Gaussian** densities
- The direct transport is usually dense, but low-dimensional structure might lie elsewhere...

Decomposable transport maps

- **Definition:** a decomposable transport is a map $T = T_1 \circ \cdots \circ T_k$ that factorizes as the composition of finitely many maps of low effective dimension and that are triangular (up to a permutation), e.g.,

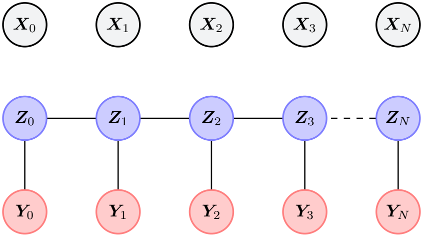
$$T(\mathbf{x}) = \underbrace{\begin{bmatrix} A_1(\mathbf{x}_1, \mathbf{x}_2) \\ B_1(\mathbf{x}_2) \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \\ \vdots \\ \mathbf{x}_n \end{bmatrix}}_{T_1} \circ \underbrace{\begin{bmatrix} \mathbf{x}_1 \\ A_2(\mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_5) \\ B_2(\mathbf{x}_3, \mathbf{x}_5) \\ \mathbf{x}_4 \\ C_2(\mathbf{x}_5) \\ \vdots \\ \mathbf{x}_n \end{bmatrix}}_{T_2} \circ \cdots \circ \underbrace{\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ A_k(\mathbf{x}_4) \\ \mathbf{x}_5 \\ \vdots \\ B_k(\mathbf{x}_4, \mathbf{x}_n) \end{bmatrix}}_{T_k}$$

- **Theorem:**[2] Decomposable graphical models for π lead to decomposable direct maps T, provided that $\eta(\mathbf{x}) = \prod_i \eta(x_i)$

[2]Spantini et al. (2017)

Applications to Bayesian filtering/smoothing

- Sparsity/decomposability apply to **general** Markov structures
- **Special case:** nonlinear non-Gaussian state-space models



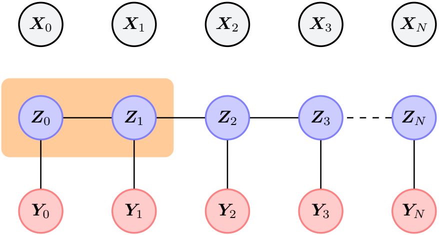
- Ideally, interested in recursively updating the full Bayesian solution:

$$\pi_{Z_{0:k} \mid Y_{0:k}} \to \pi_{Z_{0:k+1} \mid Y_{0:k+1}}$$

- Let $\mathbf{X}_0, \mathbf{X}_1, \ldots$ be an independent process with marginals $(\eta_{X_k})_k$
- Coupling between $\mathbf{X}_0, \ldots, \mathbf{X}_N$ and $\mathbf{Z}_0, \ldots, \mathbf{Z}_N | \mathbf{Y}_0, \ldots, \mathbf{Y}_N$

Seek a decomposable transport for $\pi_{Z_0, \ldots, Z_k \mid Y_0, \ldots, Y_k}$ (just a chain!)

First step: compute a 2-D map



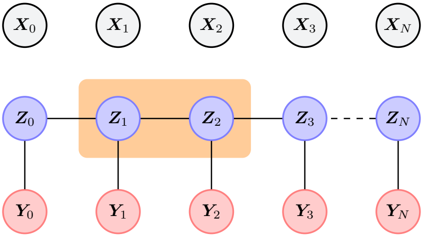
- Compute $\mathfrak{M}_0 : \mathbb{R}^{2n} \to \mathbb{R}^{2n}$ s.t.

$$\mathfrak{M}_0(\mathbf{x}_0, \mathbf{x}_1) = \begin{bmatrix} A_0(\mathbf{x}_0, \mathbf{x}_1) \\ B_0(\mathbf{x}_1) \end{bmatrix}$$

- Reference: $\eta_{X_0} \eta_{X_1}$
- Target: $\pi_{Z_0} \, \pi_{Z_1|Z_0} \, \pi_{Y_0|Z_0} \, \pi_{Y_1|Z_1}$
- $\dim(\mathfrak{M}_0) \simeq 2 \times \dim(\mathbf{Z}_0)$

$$T_0(\mathbf{x}) = \begin{bmatrix} A_0(\mathbf{x}_0, \mathbf{x}_1) \\ B_0(\mathbf{x}_1) \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}$$

Second step: compute a 2-D map



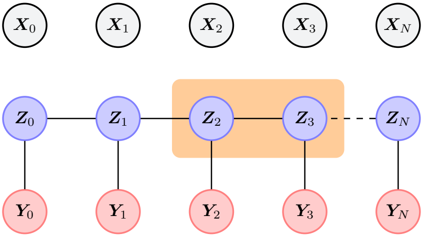
- Compute $\mathfrak{M}_1 : \mathbb{R}^{2n} \to \mathbb{R}^{2n}$ s.t.

$$\mathfrak{M}_1(\mathbf{x}_1, \mathbf{x}_2) = \begin{bmatrix} A_1(\mathbf{x}_1, \mathbf{x}_2) \\ B_1(\mathbf{x}_2) \end{bmatrix}$$

- Reference: $\eta_{X_1} \eta_{X_2}$
- Target: $\eta_{X_1}\, \pi_{Y_2|Z_2}\, \pi_{Z_2|Z_1}(\cdot|\, B_0\,(\cdot)\,)$
- Uses only one component of $\mathfrak{M}_0$

$$T_1(\mathbf{x}) = \begin{bmatrix} \mathbf{x}_0 \\ A_1(\mathbf{x}_1, \mathbf{x}_2) \\ B_1(\mathbf{x}_2) \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}$$

Proceed recursively forward in time



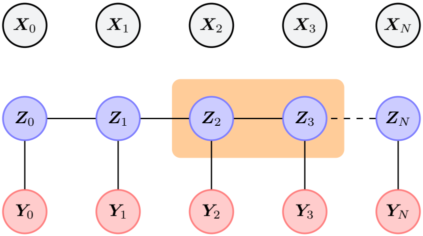
- Compute $\mathfrak{M}_2 : \mathbb{R}^{2n} \to \mathbb{R}^{2n}$ s.t.

$$\mathfrak{M}_2(\mathbf{x}_2, \mathbf{x}_3) = \begin{bmatrix} A_2(\mathbf{x}_2, \mathbf{x}_3) \\ B_2(\mathbf{x}_3) \end{bmatrix}$$

- Reference: $\eta_{X_2}\eta_{X_3}$
- Target: $\eta_{X_2}\, \pi_{Y_3|Z_3}\, \pi_{Z_3|Z_2}(\cdot|\, B_1(\cdot)\,)$
- Uses only one component of $\mathfrak{M}_1$

$$T_2(\mathbf{x}) = \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{x}_1 \\ A_2(\mathbf{x}_2, \mathbf{x}_3) \\ B_2(\mathbf{x}_3) \\ \mathbf{x}_4 \\ \mathbf{x}_5 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}$$

A decomposition theorem for chains



Theorem:[a]

1. $(B_k)_\sharp\, \eta_{X_{k+1}} = \pi_{Z_{k+1}\,|\,Y_{0:k+1}}$ (*filtering*)
2. $(\mathfrak{M}_k)_\sharp\, \eta_{X_{k:k+1}} \simeq \pi_{Z_k, Z_{k+1}\,|\,Y_{0:k+1}}$ (*lag-1 smoothing*)
3. $(T_0 \circ \cdots \circ T_k)_\sharp\, \eta_{X_{0:k+1}} = \pi_{Z_{0:k+1}|\,Y_{0:k+1}}$ (*full Bayesian solution*)

[a] Spantini et al. (2017)

A nested decomposable map

- $\mathfrak{T}_k = T_0 \circ T_1 \circ \cdots \circ T_k$ characterizes the full joint $\pi_{\mathsf{Z}_{0:k+1}|\mathsf{Y}_{0:k+1}}$

$$\mathfrak{T}_k \quad (\mathbf{x}) = \underbrace{\begin{bmatrix} A_0(\mathbf{x}_0, \mathbf{x}_1) \\ B_0(\mathbf{x}_1) \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}}_{T_0} \circ \underbrace{\begin{bmatrix} \mathbf{x}_0 \\ A_1(\mathbf{x}_1, \mathbf{x}_2) \\ B_1(\mathbf{x}_2) \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}}_{T_1} \circ$$

- Trivial to go from $\mathfrak{T}_k$ to $\mathfrak{T}_{k+1}$: just append a new map T_{k+1}
- No need to recompute $T_0, \ldots, T_k$ (nested transports)
- $\mathfrak{T}_k$ is dense and high-dimensional but **decomposable**

A nested decomposable map

- $\mathfrak{T}_k = T_0 \circ T_1 \circ \cdots \circ T_k$ characterizes the full joint $\pi_{Z_{0:k+1}|Y_{0:k+1}}$

$$\mathfrak{T}_{k+1}(\mathbf{x}) = \underbrace{\begin{bmatrix} A_0(\mathbf{x}_0, \mathbf{x}_1) \\ B_0(\mathbf{x}_1) \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}}_{T_0} \circ \underbrace{\begin{bmatrix} \mathbf{x}_0 \\ A_1(\mathbf{x}_1, \mathbf{x}_2) \\ B_1(\mathbf{x}_2) \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}}_{T_1} \circ \underbrace{\begin{bmatrix} \mathbf{x}_0 \\ \mathbf{x}_1 \\ A_2(\mathbf{x}_2, \mathbf{x}_3) \\ B_2(\mathbf{x}_3) \\ \mathbf{x}_4 \\ \mathbf{x}_5 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}}_{T_2} \circ \cdots$$

- Trivial to go from $\mathfrak{T}_k$ to $\mathfrak{T}_{k+1}$: just append a new map T_{k+1}
- No need to recompute $T_0, \ldots, T_k$ (nested transports)
- $\mathfrak{T}_k$ is dense and high-dimensional but **decomposable**

A single-pass algorithm on the model

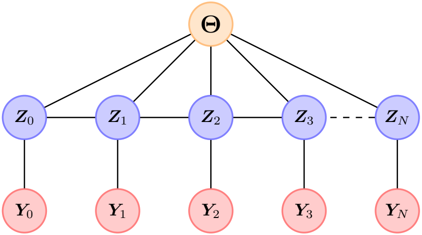
- **Meta-algorithm:**
  1. Compute the maps $\mathfrak{M}_0, \mathfrak{M}_1, \ldots$, each of dimension $2 \times \dim(\mathbf{Z}_0)$
  2. Embed each $\mathfrak{M}_j$ into an identity function to form T_j
  3. Evaluate $T_0 \circ \cdots \circ T_k$ for the full Bayesian solution
- **Remarks**:
  - A single pass on the state-space model
  - **Non-Gaussian** generalization of the Rauch-Tung-Striebel smoother
  - Bias is *only* due to the numerical approximation of each map T_i
  - Can either accept the bias or reduce it by:
    - Increasing the complexity of each map T_i, or
    - Computing **weights** given by the proposal density

$$(T_0 \circ T_1 \circ \cdots \circ T_k)_\sharp \, \eta_{\mathsf{X}_{0:k+1}}$$

  - The cost of evaluating weights grows linearly with time

Joint parameter/state estimation

- Can be generalized to sequential **joint parameter/state estimation**



- $(T_0 \circ \cdots \circ T_k)_\sharp \, \eta_\Theta \, \eta_{X_{0:k+1}} = \pi_{\Theta, Z_{0:k+1} \mid Y_{0:k+1}}$ *(full Bayesian solution)*
- However, now $\dim(\mathfrak{M}_j) = 2 \times \dim(\mathbf{Z}_j) + \dim(\Theta)$
- **Remarks**:
  - No artificial dynamic for the static parameters
  - No a priori fixed-lag smoothing approximation

Another decomposable map

$$\mathfrak{T}_{k+1}(\mathbf{x}) = \underbrace{\begin{bmatrix} P_0(x_\theta) \\ A_0(\mathbf{x}_\theta, \mathbf{x}_0, \mathbf{x}_1) \\ B_0(\mathbf{x}_\theta, \mathbf{x}_1) \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}}_{T_0} \circ \underbrace{\begin{bmatrix} P_1(x_\theta) \\ \mathbf{x}_0 \\ A_1(\mathbf{x}_\theta, \mathbf{x}_1, \mathbf{x}_2) \\ B_1(\mathbf{x}_\theta, \mathbf{x}_2) \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}}_{T_1} \circ \underbrace{\begin{bmatrix} P_2(x_\theta) \\ \mathbf{x}_0 \\ \mathbf{x}_1 \\ A_2(\mathbf{x}_\theta, \mathbf{x}_2, \mathbf{x}_3) \\ B_2(\mathbf{x}_\theta, \mathbf{x}_3) \\ \mathbf{x}_4 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}}_{T_2} \circ \cdots$$

- $(P_0 \circ \cdots \circ P_k)_\sharp \, \eta_\Theta = \pi_{\Theta \,|\, \mathsf{Y}_{0:k+1}}$ *(parameter estimation)*
- If $\mathfrak{P}_k = P_0 \circ \cdots \circ P_k$, then $\mathfrak{P}_k$ can be computed recursively as

$$\mathfrak{P}_k = \mathfrak{P}_{k-1} \circ P_k$$

via **regression** $\Longrightarrow$ cost of evaluating $\mathfrak{P}_k$ does not grow with k

Numerical example: stochastic volatility model

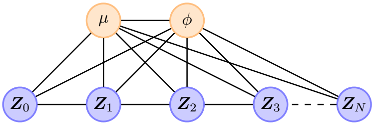
- Latent log-volatilities taking the form of an AR(1) process for $t = 0, \ldots, N$. We take $N = 944$.

$$\mathbf{Z}_{t+1} = \mu + \phi\,(\mathbf{Z}_t - \mu) + \eta_t, \quad \eta_t \sim \mathcal{N}(0,1), \quad \mathbf{Z}_0 \sim \mathcal{N}(0, 1/1-\phi^2)$$

- Observe the mean return for holding the asset at time t

$$\mathbf{Y}_t = \varepsilon_t \exp(\,0.5\,\mathbf{Z}_t\,), \quad \varepsilon_t \sim \mathcal{N}(0,1), \quad t = 0, \ldots, N$$

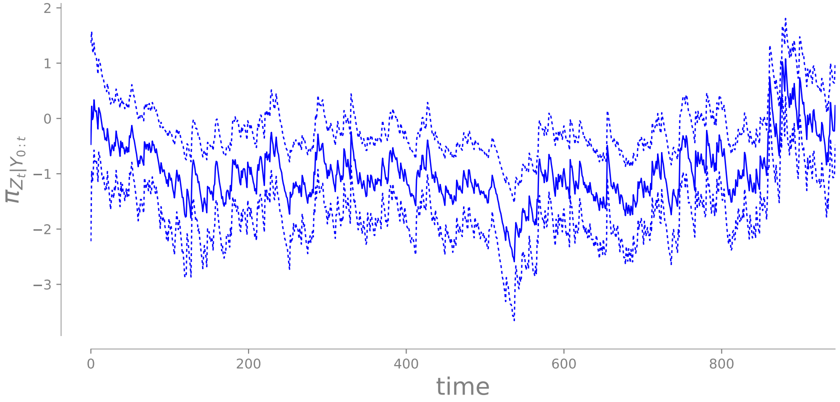
- The Markov structure for $\pi \sim \mu, \phi, \mathbf{Z}_{0:N} | \mathbf{Y}_{0:N}$ is given by:



Joint state/parameter estimation problem

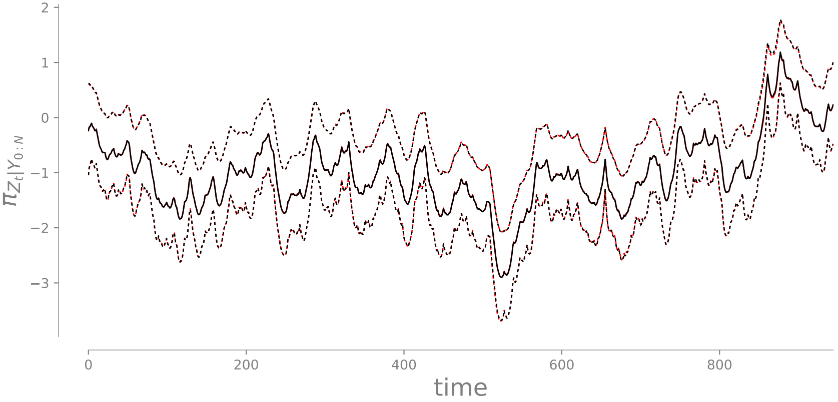
Filtering distributions

- Computed online via 4-d maps
- Can use Gauss **quadratures** for each map!



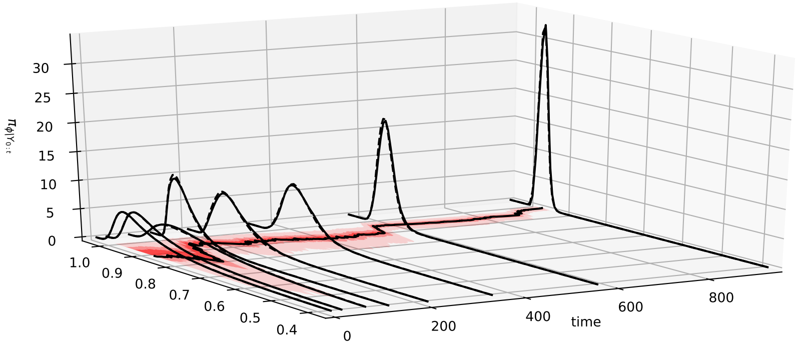
Smoothing marginals

- Just **re-evaluate** the 4-d maps backwards in time
- Comparison with a “reference” MCMC solution with 10^5 ESS (in red)



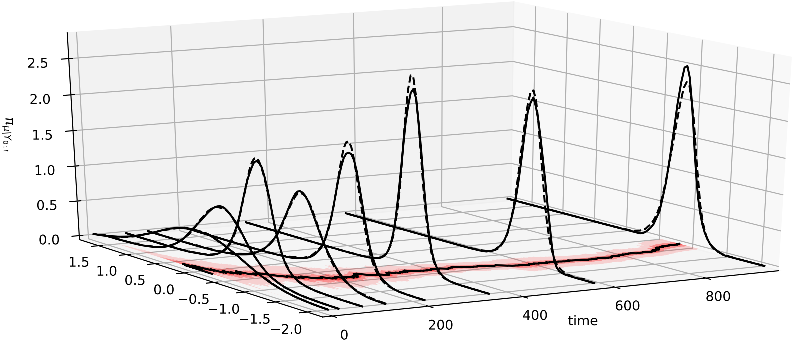
Static parameter ϕ

- **Sequential** parameter inference
- Comparison with a "reference" MCMC solution (**batch** algorithm)

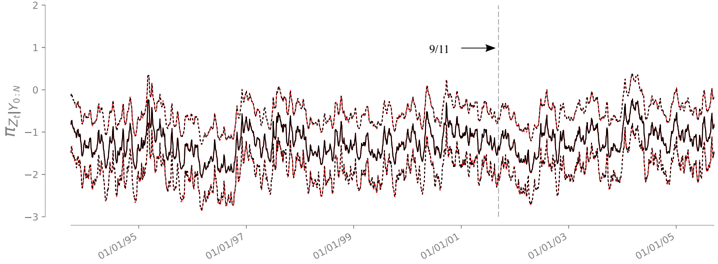


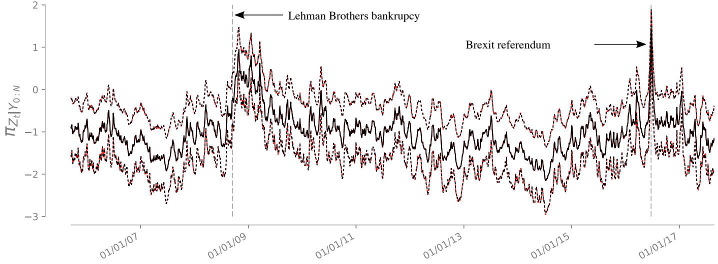
Static parameter μ

- Slow accumulation of error over time (**sequential** algorithm)
- Acceptance rate 75% for MCMC with transport-map proposal



Long-time smoothing (25 years)





- **Python code available at** `http://transportmaps.mit.edu`

Filtering high-dimensional systems

- Now we consider the filtering of state-space models with:
  1. High-dimensional states
  2. Intractable transition kernel, i.e., can only obtain forecast samples
  3. Limited model evaluations, e.g., small ensemble size
  4. Sparse and local observations in space/time
- State-of-the-art results (in terms of tracking) are *currently* obtained with localized versions of the **EnKF**
- The EnKF is not consistent, but robust

Some open questions:

- For a given ensemble size N, are we doing the best we can?
- EnKF is not guaranteed to perform better as N increases, and in some situations performs worse! Can this be mitigated?
- Can we get closer to the Bayesian solution, while preserving robustness of EnKF approaches?

Nonlinear filters induced by local couplings

Main idea

1. Propagation: apply the dynamics to obtain the next forecast ensemble
2. Assimilation: transform the forecast ensemble into approx. samples from the filtering distribution via local, nonlinear couplings

Key steps of the assimilation algorithm:

1 Approximate the *forecast distribution* on a manifold of sparse non-Gaussian Markov random fields

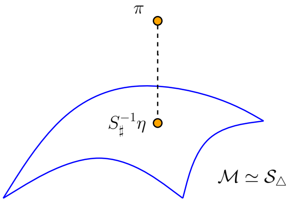
2a Local *assimilation* of the observations

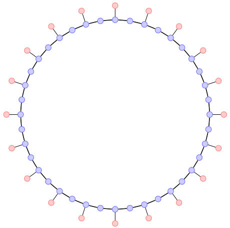
2b *Propagation* of information across the state

Abstraction of the assimilation problem:

- We have *samples* from the prior & can evaluate the likelihood

Projection onto a manifold of sparse MRFs



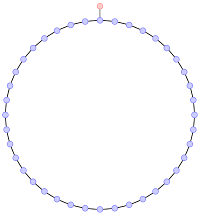


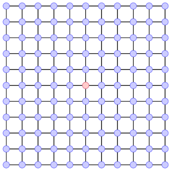
- **Approach:** learn an inverse map $S : \mathbb{R}^n \to \mathbb{R}^n$ *from samples*

$$\min_{S \in \mathcal{S}_\triangle} \mathcal{D}_{KL}(\, \pi \,||\, S_\sharp^{-1}\, \eta\,) = \max_{S \in \mathcal{S}_\triangle} \mathbb{E}_\pi[\log \eta(S(\boldsymbol{Z})) + \log |\nabla S(\boldsymbol{Z})|]$$

- Choose the approximation space $\mathcal{S}_\triangle$ (finite space of **sparse** lower triangular maps) to enforce a desired **Markov structure**
- Compute each component $S^k : \Omega \to \mathbb{R}$ via **convex** optimization
  - Choose any parameterization of S^k that departs (if desired) from linearity by adding local **nonlinear** terms (e.g., polynomials, RBFs)

Assimilation and propagation





- For simplicity, consider assimilating one observation at a time ...

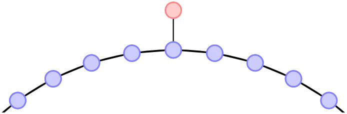
$$\pi(z|y) = \pi(z_1|y)\,\pi(z_{\sim 1}|z_1)$$

- **Local assimilation:** simulate from $\pi(z_1|y)$
  - First map component S^1 *pushes forward* the prior π_{Z_1} to η_1; yields an approximation $(S^1)^{-1}_\sharp \eta_1$ of the forecast marginal
  - Seek a direct map T^1 with target density

$$\pi(z_1|y) \propto \pi(y|z_1)\eta_1\left(S^1(z_1)\right)\partial_{z_1} S^1(z_1)$$

  - Then $T^1 \circ S^1$ transforms forecast samples of z_1 to analysis/posterior samples of z_1

Assimilation and propagation

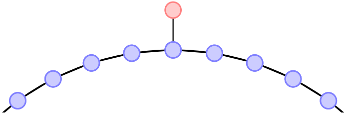


- **Propagation:** sample from the conditional $\pi(z_{\sim 1}|z_1)$ given samples from the marginal $\pi(z_1|y)$
- Given the inverse map S, notice that $S_{\boldsymbol{\xi}} : \mathbb{R}^{n-1} \to \mathbb{R}^{n-1}$,

$$z_2, \ldots, z_n \mapsto \begin{bmatrix} S^2(\boldsymbol{\xi}, z_2) \\ \vdots \\ S^n(\boldsymbol{\xi}, z_2, \ldots, z_n) \end{bmatrix},$$

  pushes forward $\pi_{\mathbf{Z}_{2:n}|z_1=\boldsymbol{\xi}}$ to $\eta_{2:n} \implies$ just invert $S_{\boldsymbol{\xi}}$
- Sparse Markov structure yields further simplifications in S, e.g.,
  1. Sparse S
  2. Parallel inversion of $S_{\boldsymbol{\xi}}$

Assimilation and propagation



Local assimilation + propagation:

- Then the **combined** map (for a *single* observation),

$$\mathcal{T}(z) = \begin{bmatrix} T^1(z_1) \\ S^{-1}_{T^1(z_1)}(z_2, \ldots, z_n) \end{bmatrix} \circ S(z),$$

  transforms the forecast ensemble to the analysis ensemble!

- Can *iterate* this construction to assimilate each additional observation, or generalize to multiple/batch observations

Lorenz 96 (40-dimensional state)

- A *hard* test-case configuration:[3]

$$\frac{\mathrm{d}\mathbf{Z}_j}{\mathrm{d}t} = (\mathbf{Z}_{j+1} - \mathbf{Z}_{j-2})\,\mathbf{Z}_{j-1} - \mathbf{Z}_j + F, \qquad j = 1, \dots, 40$$
$$\mathbf{Y}_j = \mathbf{Z}_j + \varepsilon_j, \qquad j = 1, 3, 5 \dots, 39$$

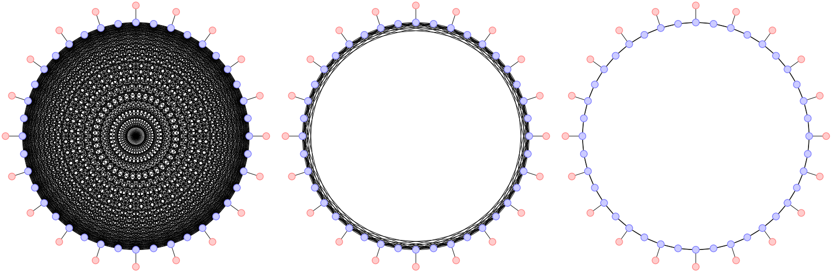
  - $F = 8$ (chaotic regime) and $\varepsilon_j \sim \mathcal{N}(0, 0.5)$
  - Time between observations: $\Delta_{\text{obs}} = 0.4$ (**large!**)
  - Results averaged over 2000 assimilation cycles

	#particles: 400		#particles: 200	
	EnKF[6]	**LocNLF**	≈EnKF	**LocNLF**
med RMSE	0.88	**0.64**	0.91	**0.66**
avg RMSE	0.97	**0.74**	1.02	**0.79**
var RMSE	0.12	**0.06**	0.1	**0.09**

- The nonlinear filter is $\sim 25\%$ more accurate in RMSE than EnKF

[3] Bengtsson et al. (2003)

Lorenz 96: details on the filtering approximation



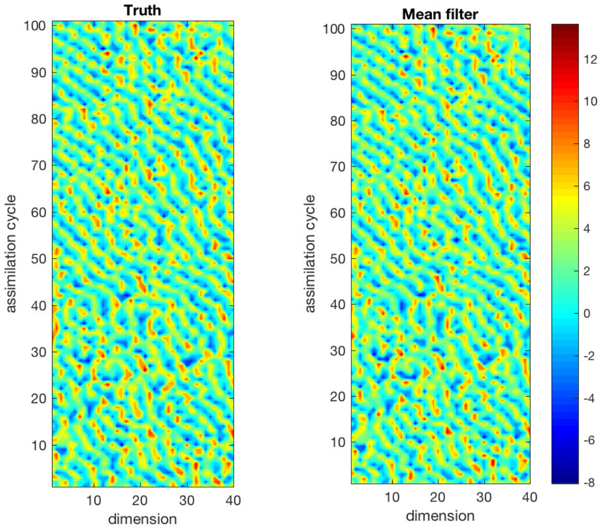
- Observations were assimilated one at a time
- **Approximate** Markov structure: 5-way interactions
- Each conditional $\pi(x_k | x_{j_1}, \ldots, x_{j_p})$ was learnt via a **separable** map

$$S^k(x_{j_1}, \ldots, x_{j_p}, x_k) = \psi(x_{j_1}) + \ldots + \psi(x_{j_p}) + \psi(x_k),$$

  where $\psi(x) = a_0 + a_1 \cdot x + \sum_{i>1} a_i \exp(-(x - c_i)^2/\sigma)$.

- Much **more general** parameterizations are of course possible!

Lorenz 96: tracking performance of the filter



- Introducing simple, **localized nonlinearities** can make a difference!

Conclusions

Summary

- Role of **continuous** transport in problems of sequential inference
  1. Filtering and smoothing (generalization of the RTS smoother)
  2. Sequential parameter-state estimation
  3. High dimensional filtering (using local couplings)

Ongoing and future work

- **Approximately sparse** Markov structures (e.g., *graph sparsification*)
- **Learn** Markov structure from samples

Thank You

Supported by the DOE Office of Advanced Scientific Computing Research

References

- A. Spantini, D. Bigoni, Y. Marzouk. "**Inference via low-dimensional couplings.**" arXiv:1703.06131 (2017)
- R. Morrison, R. Baptista, Y. Marzouk. "Beyond normality: learning sparse probabilistic graphical models in the non-Gaussian setting." NIPS 2017.
- Y. Marzouk, T. Moselhy, M. Parno, A. Spantini, "An introduction to sampling via measure transport." *Handbook of Uncertainty Quantification*, R. Ghanem, D. Higdon, H. Owhadi, eds. Springer (2016). arXiv:1602.05023.
- T. Moselhy, Y. Marzouk, "Bayesian inference with optimal maps." *J. Comp. Phys.*, 231: 7815–7850 (2012).

- **Python code at** `http://transportmaps.mit.edu`

U.S. DEPARTMENT OF ENERGY
Office of Science