Generalized Particle Filters

Dan Crisan

Imperial College London

Data Assimilation Research Centre
University of Reading

- The Filtering Problem
- Framework: discrete/continuous time
- What constitutes an approximation ?
- Quantized information = particles
- From Gaussian to particle approximations
- Generalized particle filters
- Cubature methods
- Why is the high-dimensional problem hard ?
- Space - The Final Frontier
- Stability for the high-dimensional problem
- Application to the observed Navier-Stokes equation
- Final remarks

Joint work with Alex Beskos (UCL), Ajay Jasra(NUS), Nick Kantas (UCL), Kai Li (Imperial), Salvador Ortiz-Latorre (Oslo), Joaquin Miguez (Madrid).

- X the signal process - “hidden component”
- Y the observation process - “the data”

The filtering problem: Find the conditional distribution of the *signal* X_t given $\mathcal{Y}_t = \sigma(Y_s, s \in [0,t])$, i.e.,

$$\pi_t(A) = \mathbb{P}(X_t \in A | \mathcal{Y}_t), \qquad t \geq 0, \qquad A \in \mathcal{B}(\mathbb{R}^d).$$

Continuous framework:

$$\begin{aligned} dX_t &= f(X_t)dt + \sigma(X_t)dV_t, \\ dY_t &= h(X_t)dt + dW_t. \end{aligned}$$

Discrete framework:

$$\begin{aligned} &\{X_t\}_{t\geq 0} \text{ Markov chain } \mathbb{P}(X_t \in A | X_{t-1} = x_{t-1}) = \tau_t(A|x_{t-1}), \\ &\{X_t, Y_t\}_{t\geq 0} \quad \mathbb{P}(Y_t \in dy | X_t = x_t) = g_t(y|x_t)dy \end{aligned}$$

The description of a numerical approximation for the solution of the filtering problem should contain three parts:

1. The method of recording the approximation:

particle approximations	Gaussian approximations
$(\underbrace{a_j(t)}_{\text{weight}}, \underbrace{v_j^1(t), \ldots, v_j^d(t)}_{\text{position}})_{j=1}^n$	$(\underbrace{a_j(t)}_{\text{weight}}, \underbrace{v_j^1(t), \ldots, v_j^d(t)}_{\text{mean}}, \underbrace{\omega_j^{11}(t), \ldots, \omega_j^{dd}(t)}_{\text{covariance matrix}})_{j=1}^n$
$\pi_t \rightsquigarrow \pi_t^n = \sum_{j=1}^n a_j(t)\, \delta_{v_j(t)}$	$\pi_t \rightsquigarrow \pi_t^n = \sum_{j=1}^n a_j(t)\, N(v_j(t), \omega_j(t))$

2. The law of evolution of the approximation:

particle approximations	Gaussian approximations
$\pi_t^n \overset{\text{mutation}}{\underset{\text{model}}{\longrightarrow}} \bar{\pi}_{t+\delta}^n \overset{\text{selection}}{\underset{\{Y_s\}_{s\in[t,t+\delta]}}{\longrightarrow}} \pi_{t+\delta}^n$	$\pi_t^n \overset{\text{forecast}}{\underset{\text{model}}{\longrightarrow}} \bar{\pi}_{t+\delta}^n \overset{\text{assimilation}}{\underset{\{Y_s\}_{s\in[t,t+\delta]}}{\longrightarrow}} \pi_{t+\delta}^n$

3. The measure of the approximating error:

$$\sup_{\{\varphi\in C_b,\ \ \|\varphi\|\leq 1\}} \mathbb{E}\left[|\pi_t^n(\varphi) - \pi_t(\varphi)|\right], \quad \hat{\pi}_t - \hat{\pi}_t^n, \quad \|\pi_t^n - \pi_t\|_{TV}.$$

The quantized information is modelled by n stochastic processes

$$\{p_i(t), t > 0\} \;\; i = 1, ..., n, \;\; p_i(t) \in \mathbb{R}^N.$$

- We think of the processes p_i as the trajectories of n (generalized) particles.
- Typically $N > d$, where d is the dimension of the state space.

$$\pi_t^n = \Lambda_t^n(p_i(t), t > 0 \;\; i = 1, ..., n).$$

- Generalized particle filters:
  - classical particle filters
  - gaussian approximations
  - wavelets
  - grid methods

Equidistant partition of the interval $[0, T]$ with mesh size δ.

Algorithm for $\pi_t^n = \sum_{j=1}^n \bar{a}_j(t) N(v_j(t), \omega_j(t))$.

Step 1: Initialisation

- Weight, mean, variance $(a_j(0) = 1/n,\ v_j(0) \sim \pi_0,\ \omega_j(0) = \alpha),\ \ \alpha \propto 1/\sqrt{n}$.

Step 2: Prediction/Forecast $[(i-1)\delta, i\delta)$

$$dv_j(t) = f(v_j(t))dt + \sqrt{1-\alpha}\sigma(v_j(t))dV_t^{(j)},$$
$$d\omega_j(t) = \alpha\sigma^2(v_j(t))dt,\ \ \omega_j((i-1)\delta) = \alpha$$

Step 3: Correction/Assimilation (at $i\delta$):

$$da_j(t) = a_j(t)h(v_j(t))dY_t,\ \ a_j((i-1)\delta) = 1$$
$$\bar{a}_j(t) = \frac{a_j(t)}{\sum_{i=1}^n a_i(t))}.$$

- Replace $N(v_j(i\delta-), \omega_j(i\delta-))$ with $o_j(i\delta)$ particles $N(v_j(i\delta), \alpha)$. (use TBBA)

$$v_j(i\delta) \sim N(v_j(i\delta-), \omega_j(i\delta-)),\ \ \mathbb{E}[o_j(i\delta)] = n\bar{a}_j^n(t).$$

- Re-index the positions of the particles $N(v_j(i\delta), \omega_j(i\delta))$.
- Re-initialize $a_j(i\delta) = 1$.

Theorem (DC-Li (2012))

$$\sup_{\{\varphi \in C_b^m,\ \|\varphi\| \leq 1\}} \mathbb{E}[|\pi_t^n(\varphi) - \pi_t(\varphi)|] \leq \frac{c}{\sqrt{n}}.$$

Theorem (DC-Li (2012))

The process $\sqrt{n}(\pi_t^n - \pi_t)$ converges in distribution to a process $\bar{U}$.

$$\pi_t^n \sim \pi_t + \frac{\bar{U}}{\sqrt{n}}$$

$$dX_t = f(X_t)dt + dV_t,$$
$$dY_t = h(X_t)ds + dW_t,$$

$f(x) = 0.3\tanh(0.3x)$, W and V standard Bm, $h(x) = 0.8x$.
$X_0 = 0.0,\ T = 10.0$.

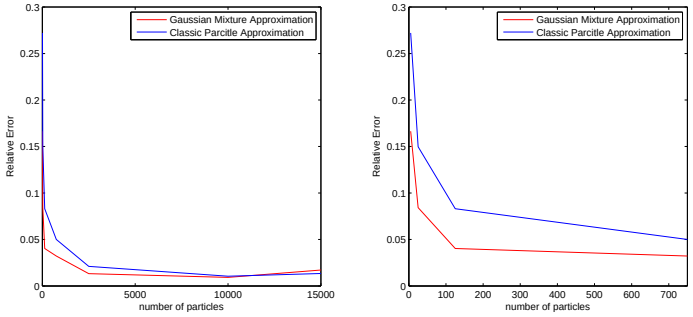


Figure: $|\pi_T^n(\varphi) - \pi_T(\varphi)| / |\pi_T(\varphi)|$ for $\varphi(x) = x^2$

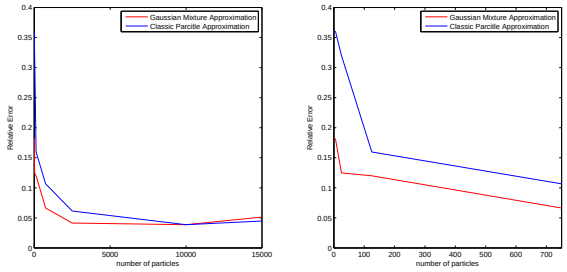
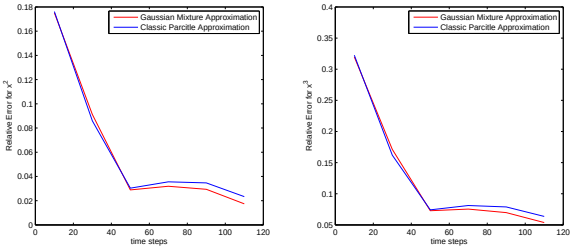


Figure: $|\pi_T^n(\varphi) - \pi_T(\varphi)| / |\pi_T(\varphi)|$ for $\varphi(x) = x^3$



The Kallianpur-Striebel formula

$$\pi_t(\varphi) = \frac{\rho_t(\varphi)}{\rho_t(\mathbf{1})},$$

where

$$\begin{aligned}\rho_t(\varphi) &= \mathbb{E}\left[\varphi(X_t)\exp\left(\int_0^t \sum_{k=1}^m h_k(X_s)\,dY_s^k - \frac{1}{2}\int_0^t \sum_{k=1}^m h_k(X_s)^2\,ds\right)\middle|\,\mathcal{Y}_t\right] \\ &= \mathbb{E}[\Lambda_t(V,Y)] = \int_{\omega\in C([0,\infty),\mathbb{R}^d)} \Lambda_t(\omega, Y)dP_V(\omega)\end{aligned}$$

A three-step scheme:

- approximate $\Lambda_{t,x}$ with an explicit/simple version $\tilde{\Lambda}_t$
- replace P_V with $P_{\tilde{V}} = \frac{1}{n}\sum_{i=1}^n \delta_{\omega_i}$ - $\tilde{V}$ approximates the signature of V
- control the computational effort (use the TBBA)

$$\rho_t(\varphi) \simeq \frac{1}{n}\sum_{i=1}^n \tilde{\Lambda}_{t,x}(\omega_i)$$

Theorem (DC, Ortiz-Latorre 2011)

$$\mathbb{E}[|\pi_t^{\delta,n}(\varphi) - \pi_t(\varphi)|] \leq C\left(\delta^{\alpha} + \delta^{\frac{m-1}{2}} + \frac{1}{\sqrt{n}}\right).$$

- The cubature method is essentially *deterministic*. The diffusion approximation uses a set of ordinary differential equations to approximate the distribution of the solution of the SDE.
- The (exponentially) increase in the computational effort is controlled by the TBBA (a random method).

Consider the 1-dimensional Benes filter:

$$\begin{aligned} dX_t &= \mu\sigma \tanh\left(\frac{\mu X_t}{\sigma}\right) dt + \sigma dV_t \\ dY_t &= (h_1 X_t + h_2)dt + dU_t, \end{aligned}$$

Then

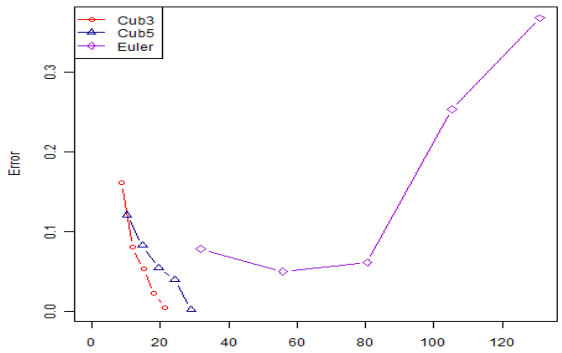
$$\rho_t \simeq w^+ \mathcal{N}(A_t^+/(2B_t), 1/(2B_t)) + w^- \mathcal{N}(A_t^-/(2B_t), 1/(2B_t)),$$

where

$$\begin{aligned} w_t^{\pm} &\triangleq \exp\left((A_t^{\pm})^2/(4B_t)\right) / (\exp\left((A_t^+)^2/(4B_t)\right) + \exp\left((A_t^-)^2/(4B_t)\right)) \\ A_t^{\pm} &\triangleq \pm\frac{\mu}{\sigma} + h_1\Psi_t + \frac{h_2 + h_1 x_0}{\sigma \sinh(h_1\sigma t)} - \frac{h_2}{\sigma}\coth(h_1\sigma t), \\ B_t &\triangleq \frac{h_1}{2\sigma}\coth(h_1\sigma t), \\ \Psi_t &\triangleq \int_0^t \frac{\sinh(h_1\sigma s)}{\sinh(h_1\sigma t)} dW_s, \end{aligned}$$

Comparison with the classical particle filter implemented using the Euler scheme plus the TBBA to perform the resampling at each time step.
Number of launches $M = 10$
Number of particles used in the KLV algorithm $N = 100$
Number of particles used for the classical particle $N = 10000$
Parameter values

$$\mu = 0.05, \quad h_1 = 0.8, \quad h_2 = 0.0, \quad \sigma = 1.0, x_0 = 0.0 \quad T = 20.0.$$



High dimensional problems are harder than their low dimensional counterparts. Example:

Consider

- $\Pi_1 = \mathcal{N}((0,\ldots,0), I_d)$ (mean $(0,\ldots,0)$ and covariance matrix I_d).
- $\Pi_2 = \mathcal{N}((1,\ldots,1), I_d)$ (mean $(0,\ldots,0)$ and covariance matrix I_d).
- $d(\Pi_1,\Pi_2)_{TV} = 2\,\mathbb{P}[\,|X| \leq d/2\,]$, $X \sim N(0,1)$.
- as d increases, the two measures get further and further apart, becoming singular w.r.t. each other exponentially fast.
- it becomes increasingly harder to use standard importance sampling, to construct a sample from Π_2 by using a proposal from Π_1.

Solution: The problem of 'moving' from Π_1 to Π_2 is equivalent to that of moving from a standard normal distribution $\mathcal{N}(0,1)$ to a normal distribution $\mathcal{N}(d,1)$ (the total variation distance between $\mathcal{N}(0,1)$ and $\mathcal{N}(d,1)$ is the same as that between Π_1 and Π_2). Rather than jumping from $\mathcal{N}(0,1)$ to $\mathcal{N}(d,1)$ in one step we get there in d steps: at each step moving from $\mathcal{N}(k-1,1)$ to $\mathcal{N}(k,1)$ for index $k = 1,2,\ldots,d$. This algorithm can be immediately transferred to the corresponding multidimensional set-up.

BICKEL, P., LI, B. & BENGTSSON, T. (2008). Sharp failure rates for the bootstrap particle filter in high dimensions.

> *'Unfortunately, for truly high dimensional systems, we conjecture that the number of intermediate steps would be prohibitively large and render it practically infeasible.'*

SMC algorithms with computational cost $\mathcal{O}(Nd^2)$ are stable. They require $\mathcal{O}(d)$ intermediate steps. If one takes $\mathcal{O}(d^{1+\delta})$ steps with any $\delta > 0$, then the corresponding essential sample size (ESS) converges in probability to N and the Monte Carlo error is the same as with i.i.d. sampling. If $-1 < \delta < 0$ then ESS will go-to zero. That is, $\mathcal{O}(d)$ steps are a critical order for the stability of the algorithm.

Sequential Monte Carlo

Sample from a target distribution with density Π on $\mathbb{R}^d$ with respect to Lebesgue measure, known up to a normalizing constant. We introduce a sequence of 'bridging' densities which start from an easy to sample target and evolve toward Π:

$$\Pi_n(x) \propto \Pi(x)^{\phi_n} , \quad x \in \mathbb{R}^d , \tag{1}$$

for $0 < \phi_0 < \cdots < \phi_{n-1} < \phi_n < \cdots < \phi_p = 1$. The effect of exponentiating with the small constant ϕ_0 is that $\Pi(x)^{\phi_0}$ is much 'flatter' than Π. One can sample from the sequence of densities using an SMC sampler, which is, essentially, a Sequential Importance Resampling (SIR) algorithm or particle filter that targets the sequence of densities:

$$\widetilde{\Pi}_n(x_{1:n}) = \Pi_n(x_n) \prod_{j=1}^{n-1} L_j(x_{j+1}, x_j)$$

with domain $(\mathbb{R}^d)^n$ of dimension that increases with $n = 1, \ldots, p$; here, $\{L_n\}$ is a sequence of artificial backward Markov kernels that can, in principle, be arbitrarily selected.

Let $\{K_n\}$ be a sequence of Markov kernels of invariant density $\{\Pi_n\}$ and Υ a distribution; The backward Markov kernels L_n are chosen as follows:

$$L_n(x, x') = \frac{\Pi_{n+1}(x')K_{n+1}(x', x)}{\Pi_{n+1}(x)} .$$

The Algorithm

0. Sample $X_0^1, \ldots X_0^N$ i.i.d. from Υ and compute the weights for each particle $i \in \{1, \ldots, N\}$:

$$w_0(x_0^i) = \frac{\Pi_0(x_0^i)}{\Upsilon(x_0^i)} .$$

Set $n = 1$ and $l = 0$.

1. If $n \leq p$, for each i sample $X_n^i \mid x_{n-1}^i$ from K_n and calculate the weights

$$w_n(x_{l:n-1}^i) = \frac{\Pi_n(x_{n-1}^i)}{\Pi_{n-1}(x_{n-1}^i)} w_{n-1}(x_{l:n-2}^i)$$

with the convention $x_{0:-1}^i \equiv x_0^i$.

Calculate the Effective Sample Size (ESS):

$$\mathrm{ESS}_{(l,n)}(N) := \frac{\left(\sum_{i=1}^{N} w_n(x^i_{l:n-1})\right)^2}{\sum_{i=1}^{N} w_n(x^i_{l:n-1})^2} \ . \tag{2}$$

If $ESS_{(l,n)}(N) < a$:
resample $x^1_n, \ldots x^N_n$ according to their normalised weights

$$w_n(x^i_{l:n-1}) / \sum_{j=1}^{N} w_n(x^j_{l:n-1}) \ ; \tag{3}$$

set $l = n$;
re-initialise the weights by setting $w_n(x^i_{l:n-1}) \equiv 1$, $1 \leq i \leq N$;
let $x^1_n, \ldots x^N_n$ now denote the resampled particles.
Set $n = n + 1$.
Return to the start of Step 1.

Theorem (Beskos, DC, Jasra, 2011, (no resampling))

Under additional assumptions, for any fixed $N > 1$, $ESS_{(0,d)}(N)$ converges in distribution to

$$\varepsilon_N := \frac{[\sum_{i=1}^N e^{X_i}]^2}{\sum_{i=1}^N e^{2X_i}}$$

where $X_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma_\star^2)$.

Theorem (Beskos, DC, Jasra, 2011, (with resampling))

Under additional assumptions, for any fixed $N > 1$, any $k \in \{1, \ldots, m^* + 1\}$, times $t_{k-1} < t_k$, and $s_k(d) \in (t_{k-1}(d), t_k(d))$ any sequence converging to a point $s_k \in (t_{k-1}, t_k)$, we have that $ESS_{(t_{k-1}(d), s_k(d))}(N)$ converges in distribution to a random variable

$$\frac{[\sum_{i=1}^N e^{X_i^k}]^2}{\sum_{i=1}^N e^{2X_i^k}}$$

where $X_i^k \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2_{t_{k-1}:s_k})$ and $\sigma^2_{t_{k-1}:s_k} \leq \sigma_\star^2$.

2D Stochastic Navier-Stokes equation on the torus $\mathbb{T}^2 \triangleq [0,L) \times [0,L)$ with periodic boundary conditions:

$$\begin{aligned} \frac{\partial u}{\partial t} - \nu \Delta u + u \cdot \nabla u + \nabla p &= f + W(t,x) && \text{for all } (x,t) \in \mathbb{T}^2 \times (0,\infty), \quad (4) \\ \nabla \cdot u &= 0 && \text{for all } (x,t) \in \mathbb{T}^2 \times (0,\infty), \\ u(x,0) &= u_0(x) && \text{for all } x \in \mathbb{T}^2. \end{aligned}$$

- $u : \mathbb{T}^2 \times [0,\infty) \to \mathbb{R}^2$ - the velocity
- $p : \mathbb{T}^2 \times [0,\infty) \to \mathbb{R}^2$ - the pressure
- $f : \mathbb{T}^2 \to \mathbb{R}^2$ - the forcing
- $W(t,x)$ - noise

$$H \triangleq \overline{\left\{ L - \text{periodic trig. pol. } u : [0,L)^2 \to \mathbb{R}^2 \middle| \nabla \cdot u = 0, \int_{\mathbb{T}^2} u(x) dx = 0 \right\}}^{L^2(\mathbb{T}^2))^2}$$

$P : (L^2(\mathbb{T}^2))^2 \to H$ - the Leray-Helmholtz orthogonal projector. An orthonormal basis for H is given by

$$\psi_k(x) \triangleq \frac{k^\perp}{|k|} \exp\left(\frac{2\pi i k \cdot x}{L} \right) \quad k = (k_1, k_2)^\top \in \mathbb{Z}^2 \setminus \{0\} \quad k^\perp = (k_2, -k_1)^\top.$$

For $u \in H$.

$$u = \sum_{k \in \mathbb{Z}^2 \setminus \{0\}} u_k(t)\psi_k(x).$$

$$W(t,x) = \sum_{k \in \mathbb{Z}^2 \setminus \{0\}} \varepsilon_k \psi_t(x) W_t^k \in H$$

$\{W_t^k\}_{(t \geq 0,\ k \in \mathbb{Z}^2 \setminus \{0\})}$ i.i.d. Brownian motions and

$$\sum_{k \in \mathbb{Z}^2 \setminus \{0\}} (4\pi^2 |k|^2)^s \varepsilon_k^2 < \infty \qquad \text{for } s \in \mathbb{R},$$

and then $W(t,\cdot) \in H$.
The stochastic Navier-Stokes equation can be written as

$$\frac{du}{dt} + \nu A u + \mathcal{B}(u,u) = f + W(t,x). \tag{5}$$

- $A = -P\Delta$ is the Stokes operator
- $\mathcal{B}(u,u) = P(u \cdot \nabla u)$
- f is the original forcing projected into H.

The equations for the modes:

$$du_k(t) = \left(-\nu\lambda_k u_k(t) - \alpha_k^{l,j} \sum_{l+j=k} u_l(t)u_j(t) + f_k \right) dt + \varepsilon_k dW_t^k.$$

$$\alpha_k^{l,j} = \begin{cases} \frac{2\pi i(l_2 j_1 - l_1 j_2)(k_1 j_1 + k_2 j_2)}{L|k||l||j|} & \text{if} \quad k = l + j, \\ 0 & \text{otherwise;} \end{cases}$$

Define the projection operators $P_\lambda : H \to H$ and $Q_\lambda : H \to H$ by

$$P_\lambda u = \sum_{\substack{k \in \mathbb{Z}^2 \setminus \{0\} \\ |2\pi k|^2 < \lambda L^2}} u_k(t)\psi_k(x), \quad Q_\lambda = I - P_\lambda;$$

and consider the projected eigenvalues, we obtain the following evolution equation for the approximation of $u_k(t)$, which is denoted by $\tilde{u}_k(t)$, for each $k \in \mathbb{Z} \setminus \{0\}$ with $|2\pi k|^2 < \lambda L^2$:

$$d\tilde{u}_k(t) = \left(-\nu\lambda_k \tilde{u}_k(t) - \alpha_k^{l,j} \sum_{\Gamma} \tilde{u}_l(t)\tilde{u}_j(t) + f_k \right) dt + \varepsilon_k dW_t^k; \qquad (6)$$

where the set $\Gamma \triangleq \left\{ (l,j) \middle| l + j = k \text{ and } |2\pi l|^2 < \lambda L^2 \text{ and } |2\pi j|^2 < \lambda L^2 \right\}$.

Model parameters

• we use $k1, k2 = -32, ..., 0, ...32$ (i.e. a 64^2 grid for the discrete fourier components).
• Smoothing problem approximate $p(x_0|y_{1:5})$ where each y_i is a $4x4$ grid on the torus and

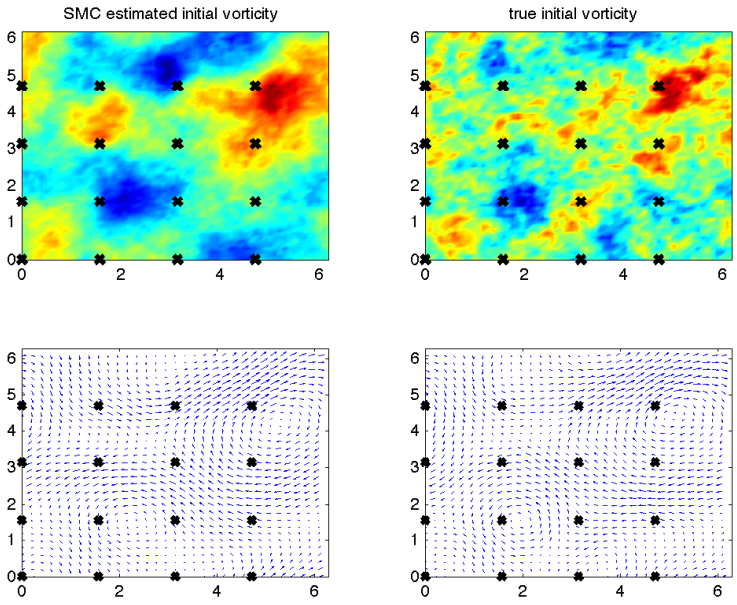
$$y_i(j) = u(x_j, t_i) + N(0, 0.2).$$

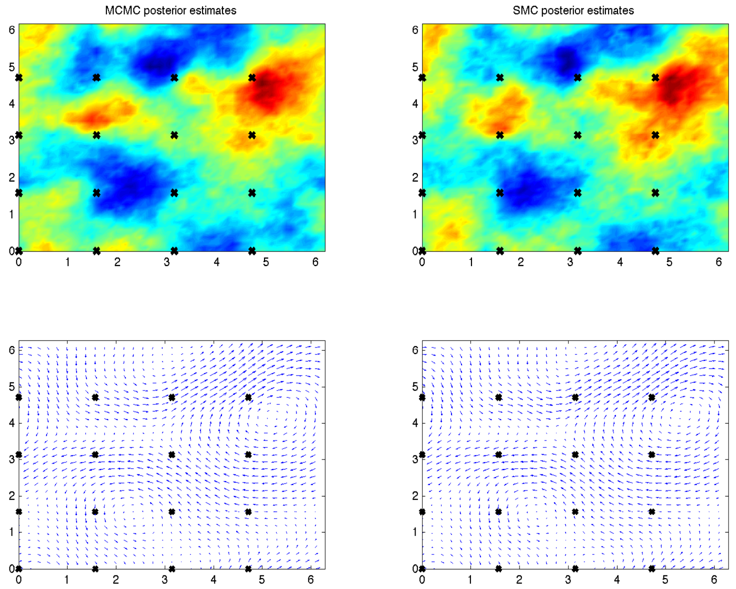
• the dynamics are initialised by a random sample from the prior $N(0, \delta A^{\alpha})$
• for the prior, $\delta = 5$ and $\alpha = 2.2$.
• torus size is 2π.
• forcing is $\nabla cos(\kappa \cdot x)$ with $\kappa = (1, 1)$ for the stationary regime and $\kappa = (5, 5)$ for the chaotic
• ν is $1/50$ for chaotic and $1/10$ for stationary

MCMC plot: computation cost roughly 10^5 iterations per day, i.e. with the slow mixing need more than 10 days for a decent but not super-reliable answer.

SMC plots: computational cost around 18 hours for N=500 particles and 5 intermediate steps.

Numerics done by N. Kantas (UCL).


SMC estimated initial vorticity
true initial vorticity


MCMC posterior estimates
SMC posterior estimates