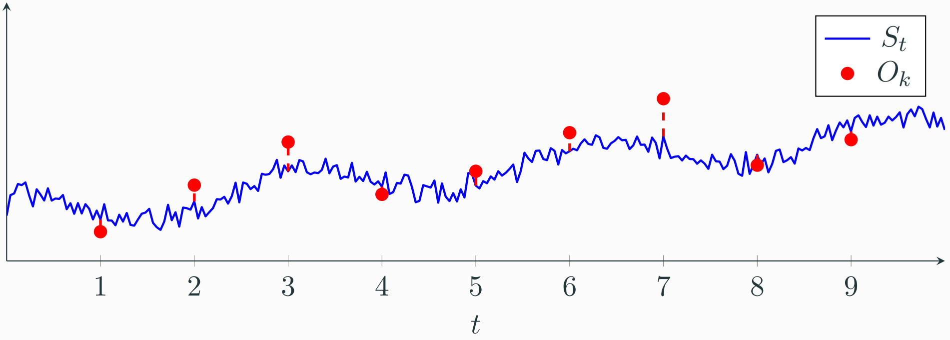
Fast Bayesian Filtering for High-Dimensional Nonlinear SDEs with Deep Density Models

Joint work with Filip Rydin

30th Nordic Conference in Mathematical Statistics, Kasper Bågmark

June 3, 2026

Chalmers University of Technology and University of Gothenburg,
Department of Mathematical Sciences



Quantity of interest

$$\mathbb{P}\left(S_{t_k} \mid O_{1:k}\right)$$

State-space model

$$S_t = S_0 + \int_0^t b(S_u)\,\mathrm{d}u + \int_0^t \sigma(S_u)\,\mathrm{d}B_u,$$
$$O_k = h(S_{t_k}) + V_k$$

Fast Bayesian filtering with deep density models

Statistical target

Nonlinear, partially observed SDEs, with the goal of approximating $p(S_{t_k} \mid O_{1:k})$

Prediction

Approximate the density evolution between observations using deep learning

Update

Incorporate new data through Bayes' rule when observations arrive

Log-density formulation

Reformulate on the log-scale, improving stability and is positivity-preserving

Numerically outperforms classical filters on a high-dimensional Lorenz–96 example

SDE

$$S_t = S_0 + \int_0^t \mu(S_r)\,\mathrm{d}r + \int_0^t \sigma(S_r)\,\mathrm{d}B_r$$

Fokker–Planck Equation

$$p(t) = p_0 + \int_0^t A^* p(s)\,\mathrm{d}s$$

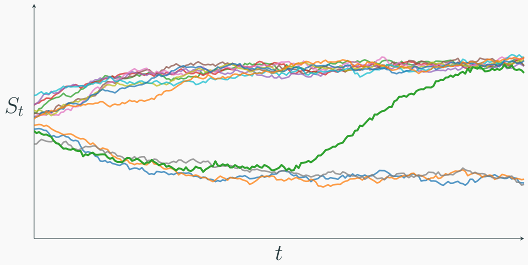
Generator

$$A\phi = \frac{1}{2}\sum_{i,j=1}^{d} a_{ij}\frac{\partial^2 \phi}{\partial x_i \partial x_j} + \sum_{i=1}^{d} \mu_i \frac{\partial \phi}{\partial x_i}$$

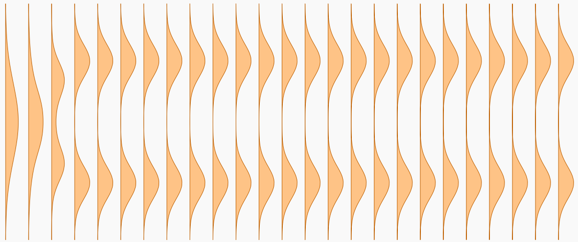
Adjoint

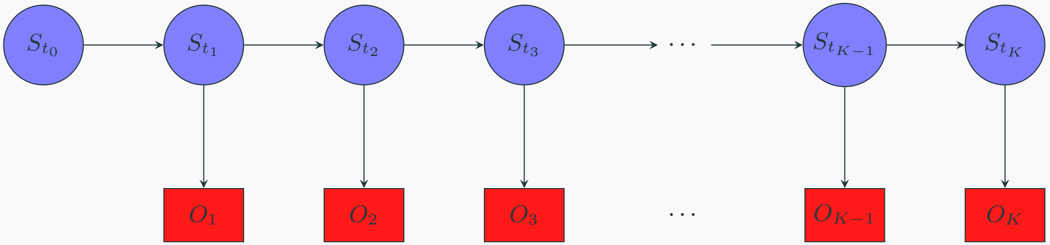
$$A^*\phi = \frac{1}{2}\sum_{i,j=1}^{d} \frac{\partial^2}{\partial x_i \partial x_j}\left(a_{ij}\phi\right) - \sum_{i=1}^{d} \frac{\partial}{\partial x_i}\left(\mu_i \phi\right)$$

Sample trajectories



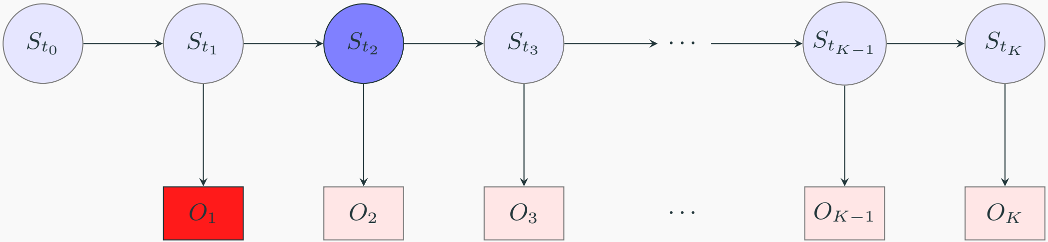
Density evolution



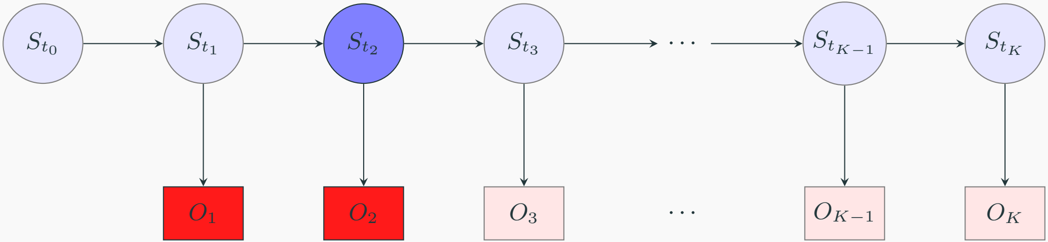


S_{t_0}
S_{t_1}
S_{t_2}
S_{t_3}
$\cdots$
$S_{t_{K-1}}$
S_{t_K}
O_1
O_2
O_3
$\cdots$
O_{K-1}
O_K

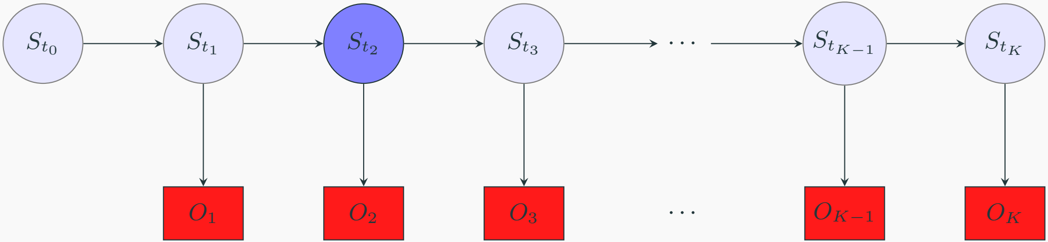
Prediction: $\mathbb{P}(S_{t_2} \mid O_1)$



Filtering: $\mathbb{P}(S_{t_2} \mid O_{1:2})$



Smoothing: $\mathbb{P}(S_{t_2} \mid O_{1:K})$



Target $p_k(t, x, o_{1:k}) := p(S_t = x \mid O_{1:k} = o_{1:k})$

(Prior) $p_0(0, x) = p(S_0 = x)$

(Prediction) $p_k(t, x, o_{1:k}) = p_k(t_k, x, o_{1:k}) + \int_{t_k}^{t} A^* p_k(s, x, o_{1:k})\,\mathrm{d}s, \quad t \in [t_k, t_{k+1}]$

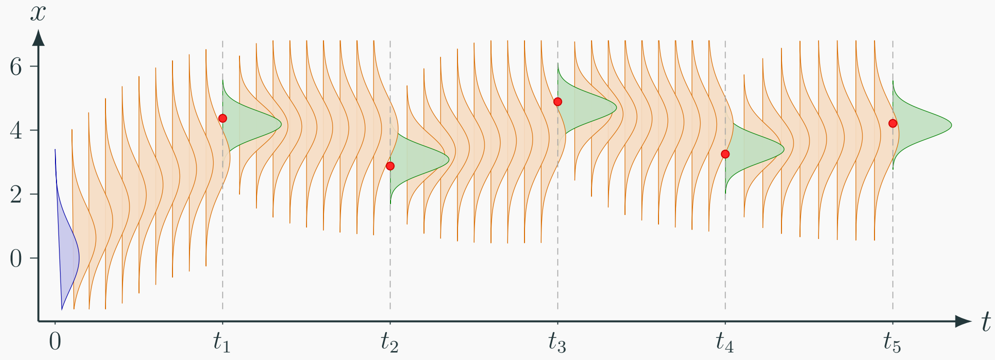
(Update) $p_{k+1}(t_{k+1}, x, o_{1:(k+1)}) = \frac{p_k(t_{k+1}, x, o_{1:k}) L(o_{k+1}, x)}{\int_{\mathbb{R}^d} p_k(t_{k+1}, z, o_{1:k}) L(o_{k+1}, z)\,\mathrm{d}z}$

Target $p_k(t, x, o_{1:k}) := p(S_t = x \mid O_{1:k} = o_{1:k})$

(Prior) $p_0(0, x) = p(S_0 = x)$

(Prediction) $p_k(t, x, o_{1:k}) = p_k(t_k, x, o_{1:k}) + \int_{t_k}^{t} A^* p_k(s, x, o_{1:k}) \, \mathrm{d}s, \quad t \in [t_k, t_{k+1}]$

(Update) $p_{k+1}(t_{k+1}, x, o_{1:(k+1)}) = \dfrac{p_k(t_{k+1}, x, o_{1:k}) L(o_{k+1}, x)}{\int_{\mathbb{R}^d} p_k(t_{k+1}, z, o_{1:k}) L(o_{k+1}, z) \, \mathrm{d}z}$



Deep Backward SDE approach

From Fokker–Planck to Backward Stochastic Differential Equation (BSDE)

Continuous-discrete filtering problem

(Prior) $p_0(0, x) = p(S_0 = x)$

(Prediction) $p_k(t, x, o_{1:k}) = p_k(t_k, x, o_{1:k}) + \int_{t_k}^{t} A^* p_k(s, x, o_{1:k}) \, \mathrm{d}s, \quad t \in [t_k, t_{k+1}]$

(Update) $p_{k+1}(t_{k+1}, x, o_{1:k+1}) = \frac{p_k(t_{k+1}, x, o_{1:k}) L(o_{k+1}, x)}{\int_{\mathbb{R}^d} p_k(t_{k+1}, z, o_{1:k}) L(o_{k+1}, z) \, \mathrm{d}z}$

From Fokker–Planck to Backward Stochastic Differential Equation (BSDE)

Focus on the prediction step

(Prior) $p_0(0, x) = p(S_0 = x)$

(Prediction) $p(t, x) = p_0(x) + \int_0^t A^* p(s, x)\,\mathrm{d}s, \quad t \in [0, T]$

(Update) $p_{k+1} \propto p_k L(o_{k+1}, \cdot)$

Quantity of interest: $p(T, x)$

From Fokker–Planck to Backward Stochastic Differential Equation (BSDE)

Focus on the prediction step

(Prior) $p_0(0, x) = p(S_0 = x)$

(Prediction) $p(t, x) = p_0(x) + \int_0^t \left[Ap(s, x) + f\big(x, p(s, x), \nabla p(s, x)\big) \right] \mathrm{d}s, \quad t \in [0, T]$

(Update) $p_{k+1} \propto p_k L(o_{k+1}, \cdot)$

Quantity of interest: $p(T, x)$

From Fokker–Planck to Backward Stochastic Differential Equation (BSDE)

Prediction step

$$p(t,x) = p_0(x) + \int_0^t A^* p(s,x)\,\mathrm{d}s, \qquad t \in [0,T]$$

Nonlinear Feynman–Kac representation

Identification

$$p(T-t, X_t) = Y_t,$$
$$\nabla p(T-t, X_t) = Z_t$$

along the forward diffusion X

BSDE representation

$$Y_t = p_0(X_T) + \int_t^T f\big(X_s, Y_s, Z_s\big)\,\mathrm{d}s - \int_t^T Z_s^\top \sigma(X_s)\,\mathrm{d}W_s, \qquad t \in [0,T]$$

The density value is represented by the backward process Y

From Fokker–Planck to Backward Stochastic Differential Equation (BSDE)

Prediction step

$$p(t,x) = p_0(x) + \int_0^t A^* p(s,x)\,\mathrm{d}s, \qquad t \in [0,T]$$

BSDE representation

$$p(T-t, X_t) = Y_t, \qquad \nabla p(T-t, X_t) = Z_t,$$

$$Y_t = p_0(X_T) + \int_t^T f_s\,\mathrm{d}s - \int_t^T Z_s^\top \sigma_s\,\mathrm{d}W_s$$

Shooting formulation

Match terminal condition

$$\min_{u \in C([0,T]\times\mathbb{R}^d;\mathbb{R})} \mathbb{E}\left[\left|u(T,X_T) - p_0(X_T)\right|^2\right]$$

Subject to the dynamics

$$u(t,X_t) = u(0,X_0) - \int_0^t f\big(X_s, u(s,X_s), \nabla u(s,X_s)\big)\,\mathrm{d}s + \int_0^t \nabla u(s,X_s)^\top \sigma(X_s)\,\mathrm{d}W_s$$

Quantity of interest: $p(T,x) = u^*(0,x)$

Shooting formulation

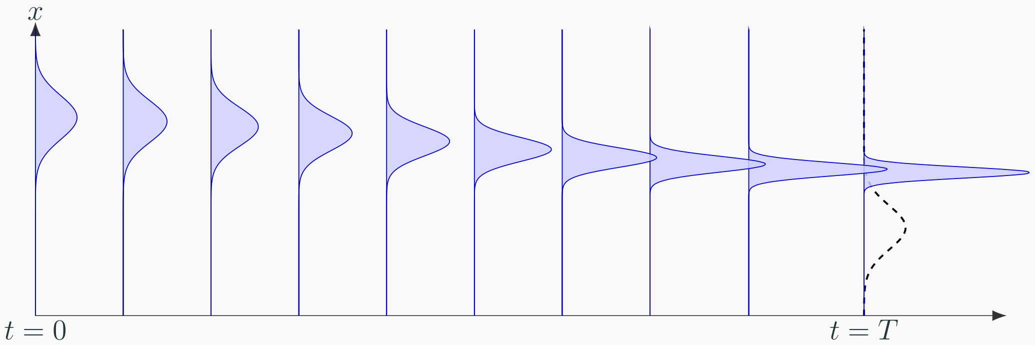
Match terminal condition

$$\min_{u \in C([0,T]\times\mathbb{R}^d;\mathbb{R})} \mathbb{E}\left[\left|u(T,X_T) - p_0(X_T)\right|^2\right]$$

Subject to the dynamics

$$u(t,X_t) = u(0,X_0) - \int_0^t f\big(X_s, u(s,X_s), \nabla u(s,X_s)\big)\,\mathrm{d}s + \int_0^t \nabla u(s,X_s)^\top \sigma(X_s)\,\mathrm{d}W_s$$

Quantity of interest: $p(T,x) = u^*(0,x)$



Shooting formulation

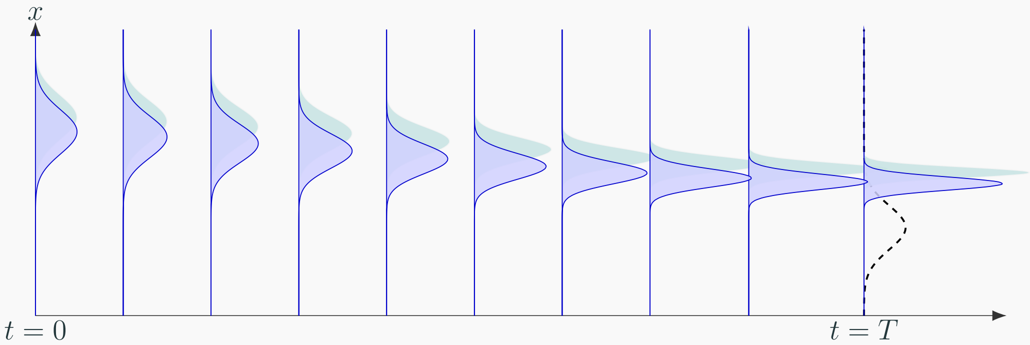
Match terminal condition

$$\min_{u \in C([0,T]\times\mathbb{R}^d;\mathbb{R})} \mathbb{E}\left[|u(T,X_T) - p_0(X_T)|^2\right]$$

Subject to the dynamics

$$u(t,X_t) = u(0,X_0) - \int_0^t f\big(X_s, u(s,X_s), \nabla u(s,X_s)\big)\,\mathrm{d}s + \int_0^t \nabla u(s,X_s)^\top \sigma(X_s)\,\mathrm{d}W_s$$

Quantity of interest: $p(T,x) = u^*(0,x)$



Shooting formulation

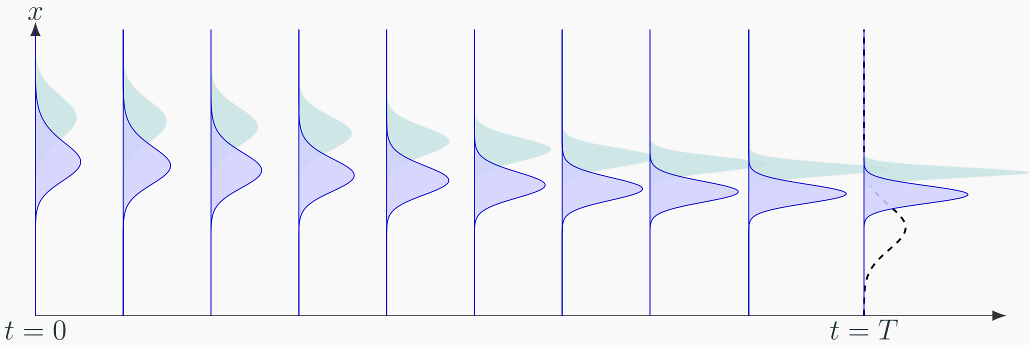
Match terminal condition

$$\min_{u \in C([0,T]\times\mathbb{R}^d;\mathbb{R})} \mathbb{E}\left[|u(T,X_T) - p_0(X_T)|^2 \right]$$

Subject to the dynamics

$$u(t,X_t) = u(0,X_0) - \int_0^t f\big(X_s, u(s,X_s), \nabla u(s,X_s)\big)\,\mathrm{d}s + \int_0^t \nabla u(s,X_s)^\top \sigma(X_s)\,\mathrm{d}W_s$$

Quantity of interest: $p(T,x) = u^*(0,x)$



Shooting formulation

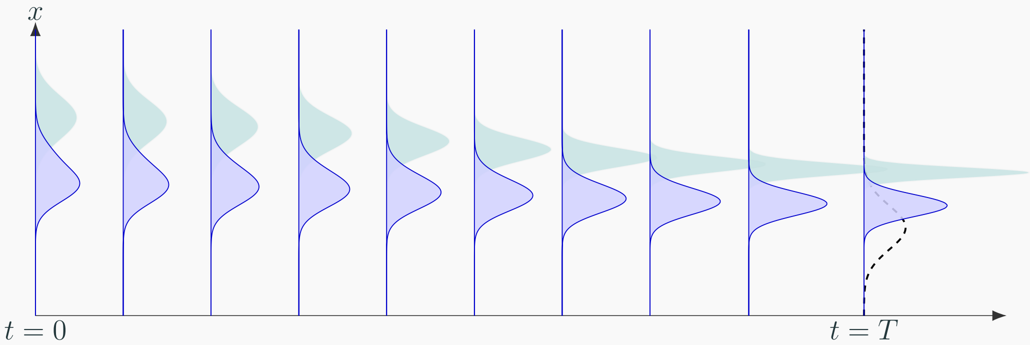
Match terminal condition

$$\min_{u \in C([0,T]\times\mathbb{R}^d;\mathbb{R})} \mathbb{E}\left[|u(T, X_T) - p_0(X_T)|^2\right]$$

Subject to the dynamics

$$u(t, X_t) = u(0, X_0) - \int_0^t f\big(X_s, u(s, X_s), \nabla u(s, X_s)\big)\,\mathrm{d}s + \int_0^t \nabla u(s, X_s)^\top \sigma(X_s)\,\mathrm{d}W_s$$

Quantity of interest: $p(T, x) = u^*(0, x)$



Shooting formulation

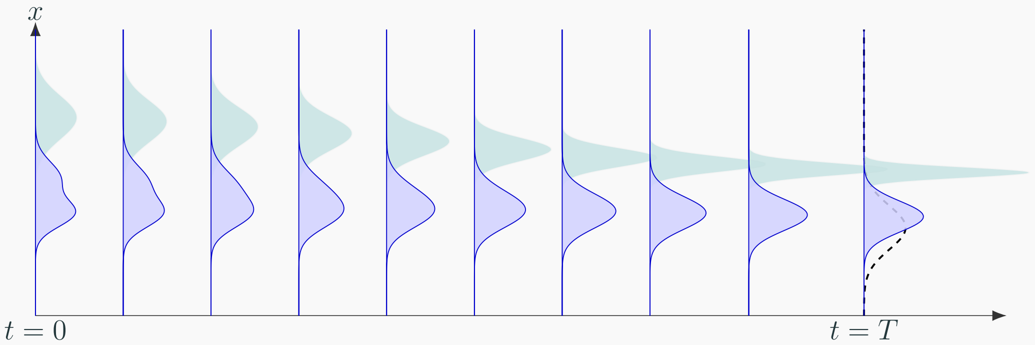
Match terminal condition

$$\min_{u \in C([0,T]\times\mathbb{R}^d;\mathbb{R})} \mathbb{E}\left[|u(T,X_T) - p_0(X_T)|^2 \right]$$

Subject to the dynamics

$$u(t,X_t) = u(0,X_0) - \int_0^t f\big(X_s, u(s,X_s), \nabla u(s,X_s)\big)\,\mathrm{d}s + \int_0^t \nabla u(s,X_s)^\top \sigma(X_s)\,\mathrm{d}W_s$$

Quantity of interest: $p(T,x) = u^*(0,x)$



Shooting formulation

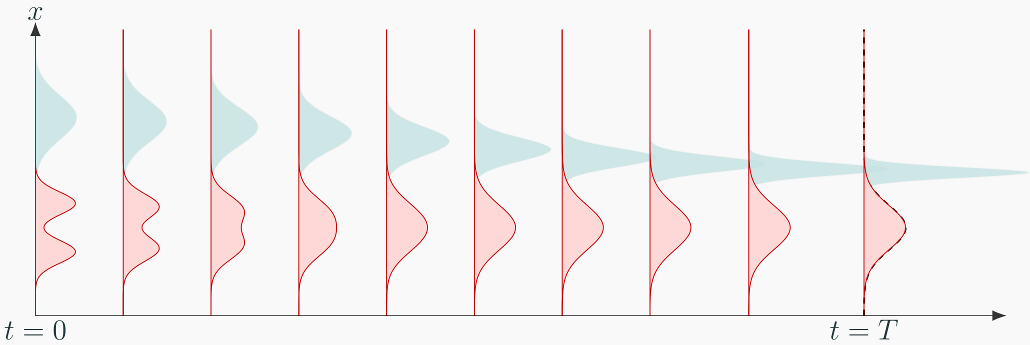
Match terminal condition

$$\min_{u \in C([0,T]\times\mathbb{R}^d;\mathbb{R})} \mathbb{E}\left[\left|u(T,X_T) - p_0(X_T)\right|^2\right]$$

Subject to the dynamics

$$u(t,X_t) = u(0,X_0) - \int_0^t f\big(X_s, u(s,X_s), \nabla u(s,X_s)\big)\,\mathrm{d}s + \int_0^t \nabla u(s,X_s)^\top \sigma(X_s)\,\mathrm{d}W_s$$

Quantity of interest: $p(T,x) = u^*(0,x)$



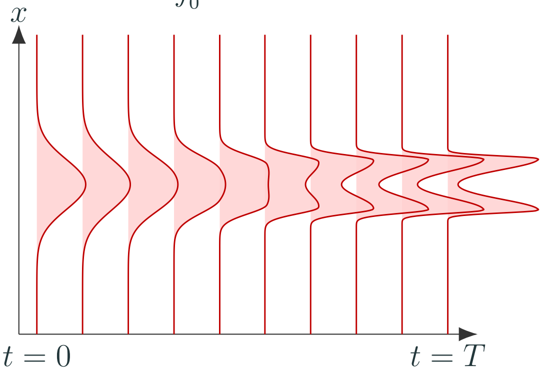
Deep splitting approach

Splitting the filtering equation

Two split evolutions

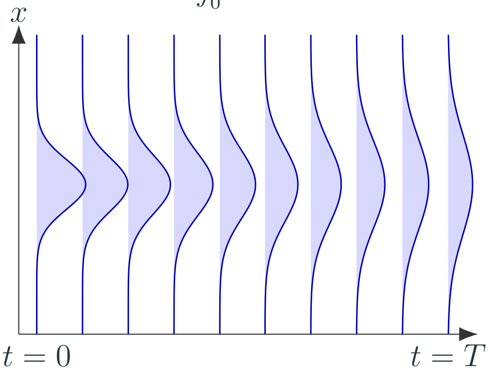
First order part

$$p_t^{(1)} = p_0 + \int_0^t f\left(p_s^{(1)}, \nabla p_s^{(1)}\right) \mathrm{d}s = \Psi_f^t p_0$$



Second order part

$$p_t^{(2)} = p_0 + \int_0^t A p_s^{(2)} \, \mathrm{d}s = \Psi_A^t p_0$$

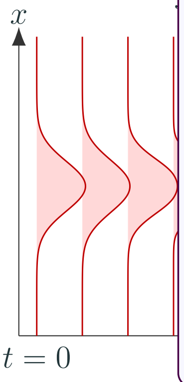


Splitting the filtering equation

Two split evolutions

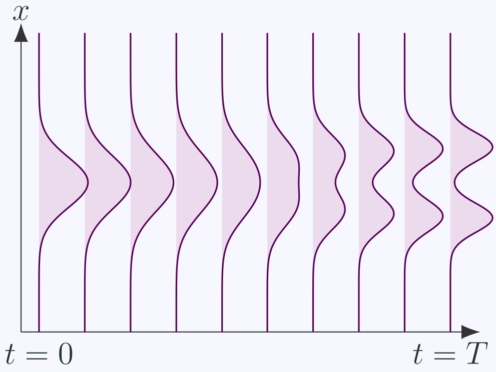
First order pa

$p_t^{(1)} = p_0 +$

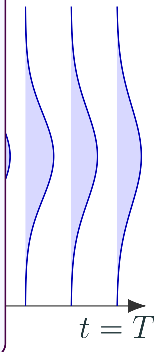


Composed evolution

$$p_{t_n} \approx \left(\Psi_A^{\Delta t} \circ \Psi_f^{\Delta t}\right)^n p_0, \qquad \Delta t = \frac{T}{N}$$



$= \Psi_A^t p_0$



Deep splitting approximation

Compose Forward Euler, Feynman–Kac, and Euler–Maruyama

Forward Euler

$$\widehat{p}^{(1)}_{t_{n+1}} = \widehat{p}^{(2)}_{t_n} + \tau f(p^{(2)}_{t_n}, \nabla p^{(2)}_{t_n})$$

Feynman–Kac

$$\widehat{p}^{(2)}_{t_{n+1}}(x) = \mathbb{E}\Big[\widehat{p}^{(1)}_{t_{n+1}}(X_{t_{n+1}}) \,\Big|\, X_{t_n} = x\Big]$$

Euler–Maruyama

$$\mathcal{X}_{n+1} = \mathcal{X}_n + b(\mathcal{X}_n)\tau + \sigma(\mathcal{X}_n)\Delta W_n$$

Composed one-step update

$$\overline{\pi}_0(x) = p_0(x)$$

$$\overline{\pi}_{n+1}(x) = \mathbb{E}\Big[\overline{\pi}_n(\mathcal{X}_{n+1}) + \tau f\big(\mathcal{X}_{n+1}, \overline{\pi}_n(\mathcal{X}_{n+1}), \nabla\overline{\pi}_n(\mathcal{X}_{n+1})\big) \,\Big|\, \mathcal{X}_n = x\Big], \quad n = 0, \dots, N-1$$

Minimization problem

$$\min_{u \in C(\mathbb{R}^d;\mathbb{R})} \mathbb{E}\Big[\Big|\overline{\pi}_n(\mathcal{X}_{n+1}) + \tau f\big(\mathcal{X}_{n+1}, \overline{\pi}_n(\mathcal{X}_{n+1}), \nabla\overline{\pi}_n(\mathcal{X}_{n+1})\big) - u(\mathcal{X}_n)\Big|^2\Big]$$

Deep splitting approximation

Compose Forward Euler, Feynman–Kac, and Euler–Maruyama

Forward Euler

$$\widehat{p}^{(1)}_{t_{n+1}} = \widehat{p}^{(2)}_{t_n} + \tau f(p^{(2)}_{t_n}, \nabla p^{(2)}_{t_n})$$

Feynman–Kac

$$\widehat{p}^{(2)}_{t_{n+1}}(x) = \mathbb{E}\Big[\widehat{p}^{(1)}_{t_{n+1}}(X_{t_{n+1}}) \,\Big|\, X_{t_n} = x\Big]$$

Euler–Maruyama

$$\mathcal{X}_{n+1} = \mathcal{X}_n + b(\mathcal{X}_n)\tau + \sigma(\mathcal{X}_n)\Delta W_n$$

Composed one-step update

$$\overline{\pi}_0(x) = p_0(x)$$

$$\overline{\pi}_{n+1}(x) = \mathbb{E}\Big[\overline{\pi}_n(\mathcal{X}_{n+1}) + \tau f\big(\mathcal{X}_{n+1}, \overline{\pi}_n(\mathcal{X}_{n+1}), \nabla\overline{\pi}_n(\mathcal{X}_{n+1})\big) \,\Big|\, \mathcal{X}_n = x\Big], \quad n = 0, \ldots, N-1$$

Minimization problem

$$\min_{u \in \mathcal{NN}(\mathbb{R}^d;\mathbb{R})} \mathbb{E}\Big[\Big|\widetilde{\pi}_n(\mathcal{X}_{n+1}) + \tau f\big(\mathcal{X}_{n+1}, \widetilde{\pi}_n(\mathcal{X}_{n+1}), \nabla\widetilde{\pi}_n(\mathcal{X}_{n+1})\big) - u(\mathcal{X}_n)\Big|^2\Big]$$

Log-density formulation

Log-density formulation

From density to log-density

(Prior) $p_0(0,x) = p(S_0 = x)$

(Prediction)
$$p_k(t,x,o_{1:k}) = p_k(t_k,x,o_{1:k}) + \int_{t_k}^{t} \Big(A p_k(s,x,o_{1:k}) + f\big(x, p_k(s,x,o_{1:k}), \nabla p_k(s,x,o_{1:k})\big)\Big)\,\mathrm{d}s$$

$$t \in [t_k, t_{k+1}]$$

(Update)
$$p_{k+1}(t_{k+1},x,o_{1:(k+1)}) = \frac{p_k(t_{k+1},x,o_{1:k})L(o_{k+1},x)}{\int_{\mathbb{R}^d} p_k(t_{k+1},z,o_{1:k})L(o_{k+1},z)\,\mathrm{d}z}$$

Define the log-density

$$v_k(t,x,o_{1:k}) = -\log p_k(t,x,o_{1:k}),$$
$$p_k(t,x,o_{1:k}) = \exp\big(-v_k(t,x,o_{1:k})\big).$$

Log-density formulation

From density to log-density

(Prior) $v_0(0, x) = -\log p(S_0 = x)$

(Prediction)
$$v_k(t, x, o_{1:k}) = v_k(t_k, x, o_{1:k}) + \int_{t_k}^{t} \Big(A v_k(s, x, o_{1:k}) + f_{\log}\big(x, v_k(s, x, o_{1:k}), \nabla v_k(s, x, o_{1:k})\big)\Big)\, \mathrm{d}s$$
$$t \in [t_k, t_{k+1}]$$

(Update)
$$v_{k+1}(t_{k+1}, x, o_{1:(k+1)}) = v_k(t_{k+1}, x, o_{1:k}) - \log\big(L(o_{k+1}, x)\big) + \log\left(\int_{\mathbb{R}^d} \exp\big(-v_k(t_{k+1}, z, o_{1:k})\big) L(o_{k+1}, z)\, \mathrm{d}z\right)$$

$$f_{\log}(x, u, w) = -\frac{1}{2}\|\sigma(x)^\top w\|^2 - f(x, 1, -w), \qquad x \in \mathbb{R}^d, \quad u \in \mathbb{R}, \quad w \in \mathbb{R}^d.$$

Numerical experiments

Accuracy

Do the methods
give accurate estimates?

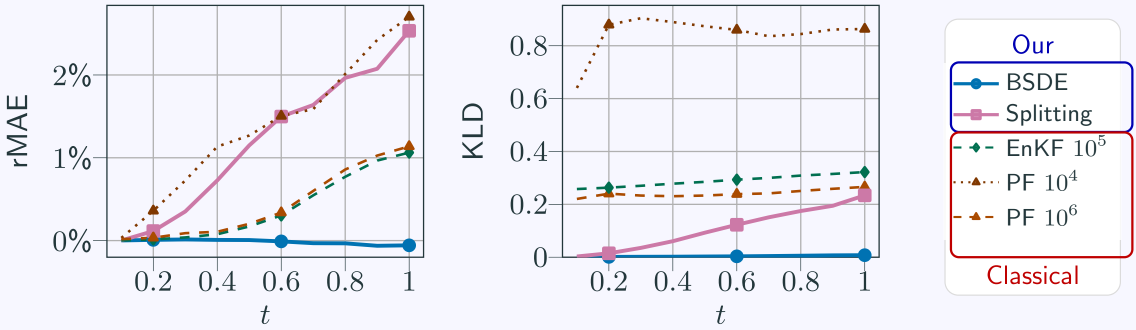
Dimension

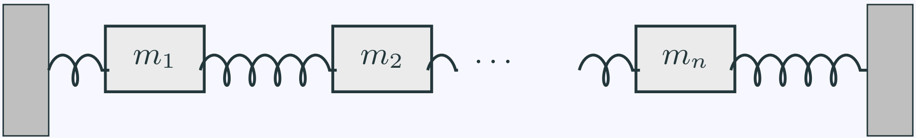
What happens as
the dimension increases?

Speed

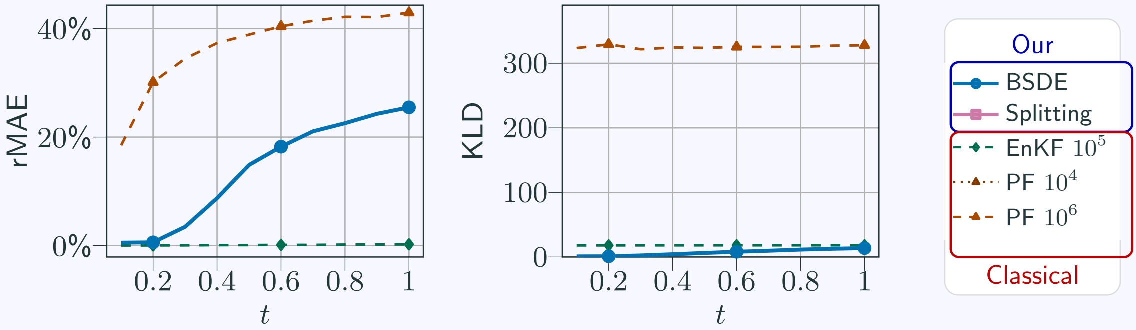
How expensive
is inference?

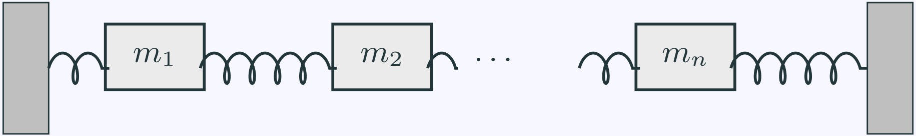
Linear spring–mass 10-dimensional





Linear spring–mass 100-dimensional

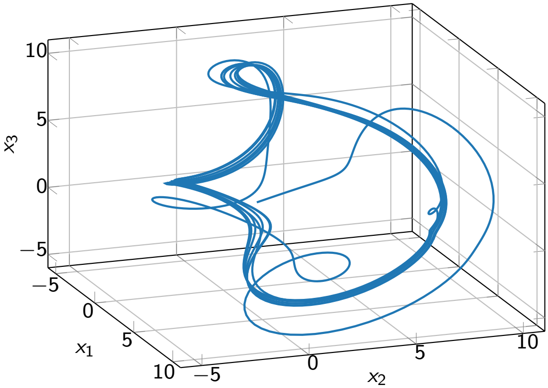




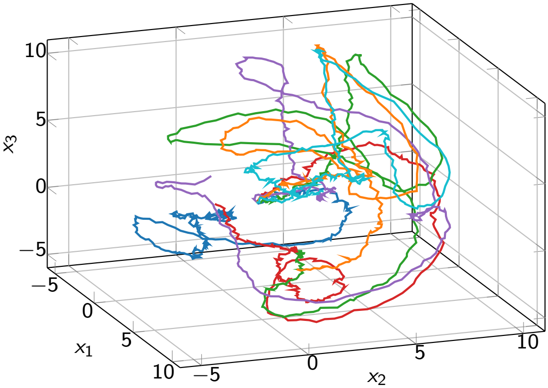
Lorenz–96 model

$$S_t^{(i)} = S_0^{(i)} + \int_0^t \left[(S_r^{(i+1)} - S_r^{(i-2)}) S_r(i-1) - S_r^{(i)} + F \right] \mathrm{d}r + \int_0^t \sigma \, \mathrm{d}W_r^{(i)}, \quad i = 1, \ldots, d$$

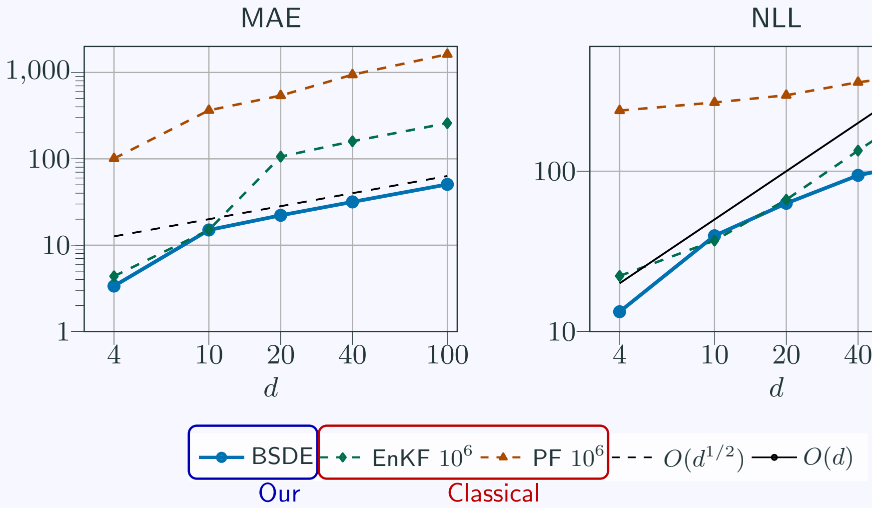
Deterministic trajectory ($\sigma = 0$)



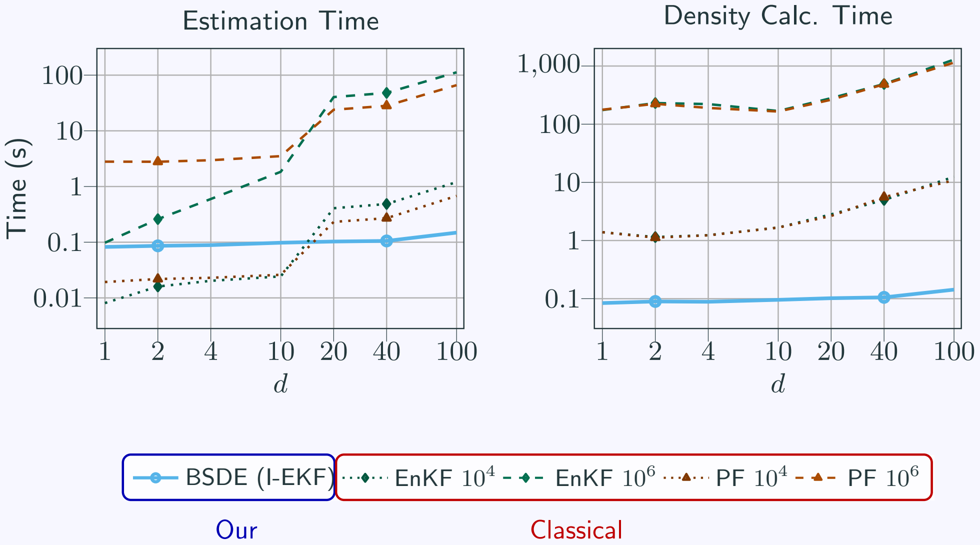
Stochastic sample paths ($\sigma = I$)



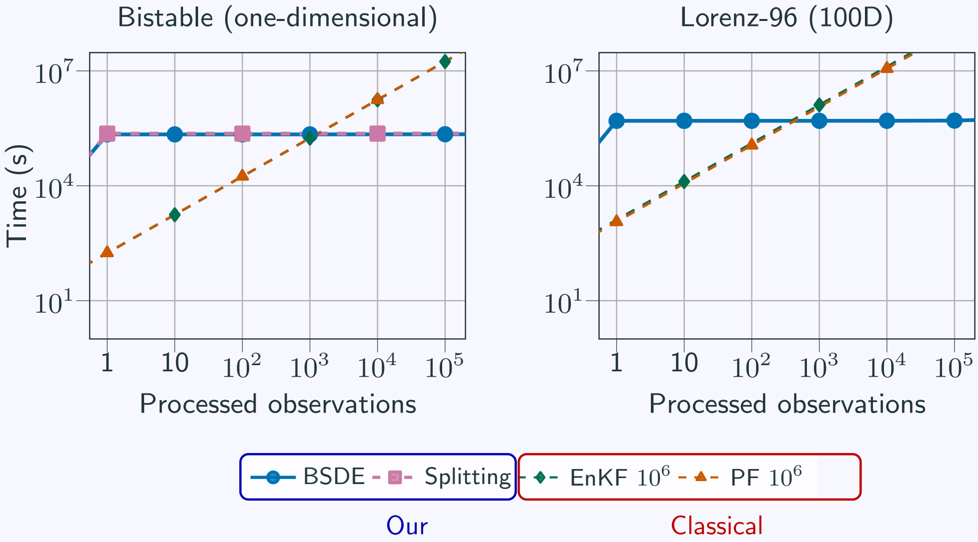
Lorenz–96 (Nonlinear system)



Computational inference time



Trade-off including training time



Conclusion

- Derived simulation-based learning approaches for state estimation
- Proved and numerically verified convergence
- The log-formulation allows for high-dimensional density targets
- Numerically, the BSDE approach outperforms the splitting approach
- The BSDE approach was successful for high-dimensional and nonlinear problems, with favorable computational time compared to classical methods

Outlook:

- Parameter inference, introduce a probability measure over a set Θ of parameters modelling μ, σ, h, p_0
- Explore other architectures

See the paper