How to avoid the curse of dimensionality: scalability of particle filters with and without importance weights

Simone Carlo Surace, Anna Kutschireiter, and Jean-Pascal Pfister

Institute of Neuroinformatics,
University of Zürich and ETH Zürich,
Zürich, Switzerland.

March 24, 2017

Abstract

Particle filters are a popular and flexible class of numerical algorithms to solve a large class of nonlinear filtering problems. However, standard particle filters with importance weights have been shown to require a sample size that increases exponentially with the dimension D of the state space in order to achieve a certain performance, which precludes their use in very high-dimensional filtering problems. Here, we focus on the dynamic aspect of this curse of dimensionality (COD) in continuous time filtering, which is caused by the degeneracy of importance weights over time. We show that the degeneracy occurs on a time-scale that decreases with increasing D. In order to soften the effects of weight degeneracy, most particle filters use particle resampling and improved proposal functions for the particle motion. We explain why neither of the two can prevent the COD in general. In order to address this fundamental problem, we investigate an existing filtering algorithm based on optimal feedback control that sidesteps the use of importance weights. We use numerical experiments to show that this Feedback Particle Filter (FPF) by Yang et al. (2013) does not exhibit a COD.

1 Introduction

Filtering problems are rarely exactly solvable with a finite amount of computational resources, requiring numerical techniques in order to approximately represent or sample from the filtering distribution. Particle filters, which have been first introduced by Gordon et al. (1993), have seen widespread use as general-purpose algorithms to solve nonlinear filtering problems (see Doucet and Johansen (2009); Künsch (2013); Crişan and Doucet (2002) for a general survey). They use sequential importance sampling in order to calculate the filtering distribution. At each time-step, samples or particles are drawn from a proposal density (such as the prior transition probability) and then re-weighted according to the observations. Particle filters have very few restrictions with regards to the type of generative model that underlies the filtering problem; they can be applied to highly nonlinear

and non-Gaussian models, which is an advantage compared to the well-known Kalman filters.

However, they suffer from a very severe problem that affects all importance sampling-based algorithms; their efficiency diminishes rapidly with increasing dimension of the state space. This 'curse of dimensionality' (COD) occurs because in high-dimensional spaces the importance weights are more likely to be degenerate, i.e. only a few weights are significant and all others are very close to zero. There have been numerous studies on the COD in particle filtering (Daum and Huang, 2003; Bengtsson et al., 2008; Bickel et al., 2008; Snyder et al., 2008), which will be described below in more detail.

In this paper, we first revisit the COD in particle filters with a focus on the dynamic aspect of the problem. We study how the time-scales of weight degeneracy and of the performance benefit due to resampling scale with dimension. We find that the time-scale of weight degeneracy is inversely proportional to the dimensionality and proportional to the logarithm of ensemble size. From this, we obtain an exponential scaling of the ensemble size that is required to obtain a fixed time-scale of weight degeneracy. We argue that because the resampling-induced benefits occur at a time-scale that scales weakly with dimension, this exponential scaling is necessary in order to avoid the collapse of the particle filter.

Next, we show that the COD can be avoided by particle filters that do not employ importance weights. Such a filter must sample the posterior distribution directly, and the particle motions have to fully take into account the observations. An unweighted particle filter for the classical filtering problem, the Feedback Particle Filter (FPF) by Yang et al. (2013), uses optimal feedback control in order to guide the particle motions according to the observations. As a result, the particles of the FPF are samples from the posterior distribution. Although the authors claim that the FPF method does not suffer from the COD, to our knowledge this has never been explicitly demonstrated. We fill this gap in the literature by demonstrating that the FPF does not suffer from the COD.

1.1 Existing literature on the COD in particle filtering

The COD of particle filters has been studied using simple back-of-the-envelope calculations (Daum and Huang, 2003; Snyder et al., 2008) and by proving statements on the convergence of the maximum weight (Bengtsson et al., 2008; Bickel et al., 2008). These works reduce the filtering problem to a single Bayesian update step. In doing so, the COD of particle filters is reduced to the general COD of importance sampling and the effects of proposal distributions or resampling on the COD are not explored in detail.

The effect of proposal distributions on particle filtering in high-dimensional problems has received some attention. In discrete time, particle filters with 'smart' proposal distributions (i.e. with improved laws that govern the motion of particles) have been shown to perform well in certain high-dimensional problems (van Leeuwen, 2009, 2010), leading some authors to conjecture that the COD could be avoided using carefully crafted proposal distributions. However, it has been argued (Snyder et al., 2015) that the system considered in van Leeuwen (2010) was effectively a low-dimensional system disguised as a high-dimensional one. A precise definition of a sequence of filtering problems of increas-

ing dimension therefore requires a notion of 'effective dimension', which also plays a role in the arguments brought forward by Daum and Huang (2003) and Snyder et al. (2008). Special properties of the model sequence, e.g. a low-dimensional dynamical system that is embedded in spaces of increasing dimensions, could potentially avoid the COD as a function of D, but not as a function of effective dimension. Improved proposals are usually designed to reduce weight degeneracy by minimizing the rate of change of the variance of the importance weights. But even the optimal proposal function is not able to completely prevent the weight degeneracy. Moreover, in continuous time it is not known whether an optimal proposal even exists.

Meanwhile, the existing literature offers only a cursory treatment of the effect of resampling on the dimensionality-dependent scaling of particle filters. While it is understood that the most widespread resampling algorithm, multinomial resampling, is suboptimal compared to branching (see Crişan and Grunwald (1998), Chopin (2004), Douc et al. (2005), Crişan (2006), and also Bain and Crişan (2009), p.250), we cannot be certain about the existence of a resampling scheme that resolves the COD. The arguments against such a possibility, which are brought forward e.g. in Snyder et al. (2015) and Chopin (2004), are mostly heuristic and can be summarized as follows: while resampling resets the particle weights to undo the effects of weight degeneracy, it does not improve the quality of the sample instantaneously. The potential improvement is temporary and comes from particle motions that originate from regions of high likelihood. In the next section we will look more closely at this mechanism and thereby support the explanation provided by Snyder et al. (2015).

2 Preliminaries

We restrict our overall discussion to the classical nonlinear filtering problem in continuous time, where the state $X_t \in \mathbb{R}^D$ and the observation $Y_t \in \mathbb{R}^D$ are D-dimensional diffusion processes that are solutions to the Itô stochastic differential equations (SDE)

$$dX_t = f(X_t)dt + g(X_t)dW_t, \tag{1}$$

$$dY_t = h(X_t)dt + dV_t, \tag{2}$$

where W_t and V_t are independent D-dimensional Brownian motions. The stochastic filtering problem is to find conditional expectations $\mathbb{E}\left[\varphi(X_t)|\mathcal{F}_t^Y\right]$, where $\varphi : \mathbb{R}^D \to \mathbb{R}$ is a measurable function and $\mathcal{F}_t^Y$ is the filtration generated by the observation process. Next, we will describe the two approximate filtering algorithms that will be studied and compared in this paper.

2.1 The Bootstrap Particle Filter (BPF)

The bootstrap or vanilla particle filter uses importance sampling (IS) to approximate the conditional expectation as

$$\mathbb{E}\left[\varphi(X_t)|\mathcal{F}_t^Y\right] \approx \bar{\varphi}_t \doteq \sum_{i=1}^{N} m_t^{(i)} \varphi(Z_t^{(i)}), \tag{3}$$

where $Z_t^{(i)} \in \mathbb{R}^D$ are the samples or 'particles' that evolve according to the dynamics of the hidden state, and $m_t^{(i)}$ are the (normalized) importance weights. This particle filter is usually formulated in discrete time (Gordon et al., 1993; Doucet and Johansen, 2009), but we will use the standard formulation of continuous-time particle filtering (see Ch. 9 of Bain and Crişan (2009) and Ch. 23 of Crişan and Rozovskii (2011)). Between resampling times, the time evolution of the particle system is given by

$$dZ_t^{(i)} = f(Z_t^{(i)})dt + g(Z_t^{(i)})dB_t^{(i)}, \tag{4}$$

$$m_t^{(i)} = \frac{M_t^{(i)}}{\sum_{j=1}^N M_t^{(j)}}, \quad dM_t^{(i)} = M_t^{(i)} h(Z_t^{(i)}) \cdot dY_t, \tag{5}$$

where $B_t^{(i)}$ are independent Brownian motions and $\cdot$ denotes the standard scalar product on $\mathbb{R}^D$. In this formulation, the unnormalized importance weight $M_t^{(i)}$ arises as a Radon-Nikodym derivative $d\mathbb{P}^{(i)}/d\tilde{\mathbb{P}}$ of the measure $\mathbb{P}^{(i)}$ under which the observations are generated from the state $Z_t^{(i)}$ to a reference measure $\tilde{\mathbb{P}}$ under which Y_t is a Brownian motion.

It is well established (Doucet and Johansen, 2009) that the BPF suffers from weight degeneracy, i.e. the weights evolve to become less equal, with only a few significant weights and all other weights being negligibly small. As a result, the variance of the IS estimate in Eq. (3) grows and performance drops. It is useful to introduce an 'effective sample size' N_{eff} that measures the number of samples that would be required to match the performance of IS with a MC sampler that samples directly from the filtering distribution. A useful approximation of effective sample size is given by the inverse of the sum of squared weights,

$$N_{\text{eff},t} \approx \left[\sum_{i=1}^N \left(m_t^{(i)}\right)^2\right]^{-1} \doteq \tilde{N}_{\text{eff},t}, \tag{6}$$

(see Martino et al. (2017) and references therein). As a measure that only depends on the importance weights and not on the locations of the samples, it is not a good predictor of performance when samples are clustered. In particular, the typical outcome of multinomial resampling from an impoverished sample is that there are only a few distinct particle positions and multiple particles at each of those positions. The new sample is still impoverished and has a large variance despite the fact that the above definition of $N_{\text{eff},t}$ yields a value of N (after resampling, all weights are reset to a value of $1/N$). With this caveat in mind, we assume that $\tilde{N}_{\text{eff},t}$ in Eq. (6) gives an upper bound on the true value of $N_{\text{eff},t}$.

Because of the weight degeneracy outlined above, the particle system has to be frequently resampled. Resampling is performed according to a schedule that is usually based either on regular resampling intervals or on the effective sample size. We use the convention that particle trajectories are right-continuous with left-side limits $Z_{t_r^-}^{(i)}$ encoding the position before resampling. At each resampling time $t = t_r$, new particle positions $(\tilde{Z}_{t_r}^{(i)})_{i=1}^n$ are drawn from the set of positions $\{Z_{t_r^-}^{(i)}, i = 1, ..., n\}$ with replacement. The

probability that $Z_{t_r^-}^{(i)}$ appears n_i times within the tuple $(\tilde{Z}_{t_r}^{(i)})_{i=1}^n$ is chosen according to the multinomial distribution,

$$\text{Prob}\,(n_1, n_2, ..., n_N) = N! \prod_{i=1}^{N} \frac{\left(m_t^{(i)}\right)^{n_i}}{n_i!}, \quad \sum_{i=1}^{N} n_i = N. \tag{7}$$

The new position at $t = t_r$ is then set to the resampled position, i.e. $Z_{t_r}^{(i)} \leftarrow \tilde{Z}_{t_r}^{(i)}$ and all weights are reset to $1/N \leftarrow \tilde{m}_{t_r}^{(i)}$. A discussion of various other forms of resampling can be found e.g. in Crişan and Grunwald (1998), Chopin (2004), Douc et al. (2005), and Crişan (2006).

2.2 A particle filter without importance weights: the Feedback Particle Filter (FPF)

In Yang et al. (2013), a particle filter was proposed that does not require importance weights, i.e. posterior expectations are approximated by unweighted averages,

$$\mathbb{E}\left[\varphi(X_t)|\mathcal{F}_t^Y\right] \approx \overline{\overline{\varphi}}_t \doteq \sum_{i=1}^{N} \varphi(Z_t^{(i)}) \tag{8}$$

(note the distinction to the weighted average $\bar{\varphi}_t$ in Eq. (3)). By definition, the time evolution of particles fully incorporates the information given by the history of observations. The particle system of the FPF evolves according to the Itô SDEs

$$\begin{aligned} dZ_t^{(i)} = &\left(f(Z_t^{(i)}) + \Omega(Z_t^{(i)}, t)\right) dt \\ &+ g(Z_t^{(i)}) dB_t^{(i)} + K(Z_t^{(i)}, t)\left[dY_t - \tfrac{1}{2}(h(Z_t^{(i)}) + \overline{\overline{h}}_t)dt\right], \end{aligned} \tag{9}$$

where K is a ($D \times D$ matrix-valued) function, and Ω is a D-dimensional vector-valued function with components given by

$$\Omega_i(z,t) = \frac{1}{2} \sum_{j=1}^{D} \sum_{k=1}^{D} K_{jk}(z,t) \frac{\partial}{\partial z_j} K_{ik}(z,t), \quad i = 1, ..., D. \tag{10}$$

The function K is the solution to an Euler-Lagrange boundary value problem (EL-BVP). The latter results from an optimal control problem: the function K is chosen such as to make the distribution of particle positions as close as possible to the posterior filtering distribution, and it ensures that the particle distribution converges to the true filtering distribution when K is chosen according to the filtering distribution. The function K, also called *gain function*, is analogous to the Kalman gain of the Kalman-Bucy filter for the linear filtering problem. As such, K (and therefore Ω) introduces interactions between the particles, in contrast to the BPF, where particles evolve independently from each other between resampling times. Another interesting aspect of the FPF is the structure of

the innovation term (Eq. (9), term in square brackets) that multiplies the gain function: the new observation dY_t is compared to the arithmetic mean of the individual particle's estimate $h(Z_t^{(i)})$ and the population estimate $\overline{\overline{h}}_t$. The function Ω comes from the conversion from Stratonovich form to Itô form.

In practical implementations of the FPF, the main difficulty lies in the solution of the EL-BVP. In multiple dimensions ($D > 1$), naïve approaches to the EL-BVP turn out to be computationally expensive. This problem has recently been addressed in Yang et al. (2016) using a Galerkin approximation of the gain function. We use the *constant gain approximation* from that paper, which corresponds to a Galerkin approximation with coordinate functions as basis functions. With this choice the gain function is

$$K_{\cdot j}(z,t) = \frac{1}{N}\sum_{i=1}^{N}\left(h_j(Z_t^{(i)}) - \overline{\overline{h}}_{t,j}\right) Z_t^{(i)}, \quad j = 1, ..., D, \tag{11}$$

which is constant in z (but not in time) and corresponds to the sample covariance matrix between the particle positions and the observation function h.

3 Results

3.1 Illustration of the curse of dimensionality for a linear problem

We consider the linear D-dimensional problem

$$dX_t = -X_t dt + \sqrt{2} dW_t, \tag{12}$$

$$dY_t = 2X_t dt + dV_t, \tag{13}$$

where the numerical factors are chosen in order to ensure a unit prior variance and time-constant and a (one-dimensional) mean squared error of the optimal filter $\mathbb{E}[(\hat{X}_{i,t} - X_{i,t})^2] = 1/2$ across all values of D. For the linear system in Eqs. (12,13), the bootstrap particle filter (BPF) is given by

$$dZ_t^{(i)} = -Z_t^{(i)} dt + \sqrt{2} dB_t^{(i)}, \tag{14}$$

$$m_t^{(i)} = \frac{M_t^{(i)}}{\sum_{j=1}^{N} M_t^{(j)}}, \quad dM_t^{(i)} = 2M_t^{(i)} Z_t^{(i)} \cdot dY_t, \tag{15}$$

where $i = 1, ..., N$ and $Z_0^{(i)} \sim \mathcal{N}(0, \mathbb{1}_{D\times D})$ and $m_0^{(i)} = 1/N$. We note that in principle all processes should carry an index that refers to the dimension D. However, in the interest of readability, we omit this index.

3.1.1 The time-scale of weight decay shortens in higher dimensions

We study the effect of the number of dimensions D on the time-scale on which the decay of $\tilde{N}_{\mathrm{eff},t}$ occurs. In Fig. 1 we show numerical results for the trial-averaged $\tilde{N}_{\mathrm{eff},t}$, illustrating the decay of $\tilde{N}_{\mathrm{eff},t}$ as a function of time for $N = 10^4$ and $D = 10, 20, ..., 50$. The decay of

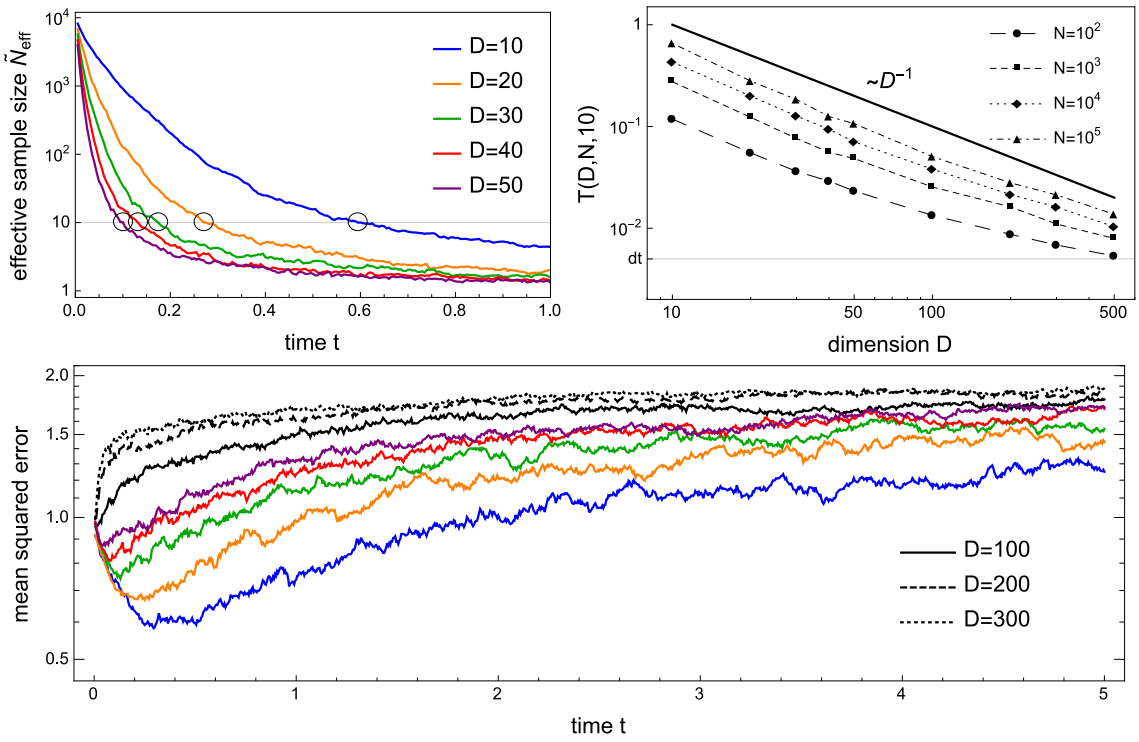

Figure 1: **Top left:** The collapse of the effective sample size $\tilde{N}_{\text{eff},t}$ over time is shown for different dimensions D of the state space and for an ensemble size $N = 10^4$. The displayed time-course is an average of 100 independent trials. **Top right:** A plot of the stopping time $T(D, N, n)$ for $n = 10$ as a function of dimension D shows an approximate D^{-1} scaling (thick black line). **Bottom:** The time evolution of the mean squared error shows a dip and a subsequent deterioration of performance due to weight decay for dimensions up to 50. The dip is shallower and the deterioration faster for higher dimensions. For even higher dimensions, the dip is no longer visible because the weight decay is too quick. All traces are averages of 100 independent trials with an ensemble size of $N = 10^4$.

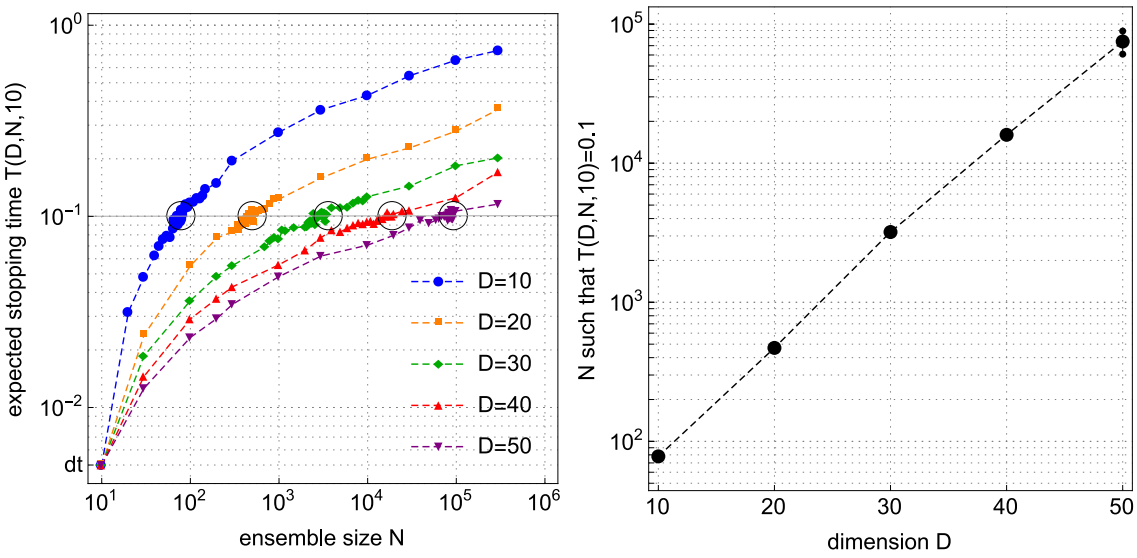

Figure 2: **Left:** The time-scale T of weight collapse (defined as the time in takes for $\tilde{N}_{\mathrm{eff},t}$ to reach a value of 10, see black circles in Fig. 1 top left) as a function of ensemble size N and dimension D. **Right:** The required ensemble size to achieve $T = 0.1$ (see black circles in the left panel) increases exponentially with dimension D.

$\tilde{N}_{\mathrm{eff},t}$ critically limits the performance of the filter. We measure the performance by the mean squared error (MSE) of the particle estimate of the hidden state

$$\mathrm{MSE}_t = \frac{1}{D}\mathbb{E}\left[\left\| X_t - \sum_{i=1}^{N} m_t^{(i)} Z_t^{(i)} \right\|^2\right], \tag{16}$$

where the average is estimated numerically by averaging independent trials, as in the case of expected stopping time. In our numerical experiment, the initial value is $\mathrm{MSE}_0 = 1$ due to the initialization of the particles according to the prior distribution. The MSE reaches an asymptotic value of two as the effective sample size goes to one. As we show in Fig. 1 bottom, for low values of D the MSE has a transient dip that is followed by a gradual increase towards the asymptotic value. The dip becomes shallower and the increase becomes faster as the dimension D increases. For very high D, the dip disappears and the MSE increases immediately.

In order to quantify the time-scale of weight decay, we define the following expected stopping time, the mean first-passage time of $\tilde{N}_{\mathrm{eff},t}$ through n,

$$T(D,N,n) = \mathbb{E}\left[\inf\left\{t \geq 0 | \tilde{N}_{\mathrm{eff},t} \leq n\right\}\right], \quad n \leq N, \tag{17}$$

as a measure for the time-scale of weight degeneracy. By definition, we have $T(D,N,N) = 0$ and $T(D,N,1) = \infty$ for all $N \in \mathbb{N}$.

We first give a rough analytical estimate of the dependence of $T(D, N, n)$ on D. The time evolution of the normalized weight can be written as

$$dm_t^{(i)} = 4m_t^{(i)}(Z_t^{(i)} - \bar{Z}_t) \cdot (X_t - \bar{Z}_t)dt + 2m_t^{(i)}(Z_t^{(i)} - \bar{Z}_t) \cdot dV_t, \tag{18}$$

for the linear model. From this we obtain

$$\mathbb{E}\left[d\left(m_t^{(i)}\right)^2\right] = \mathbb{E}\left[4\left(m_t^{(i)}\right)^2\left\{2(Z_t^{(i)} - \bar{Z}_t) \cdot (X_t - \bar{Z}_t) + \left\|Z_t^{(i)} - \bar{Z}_t\right\|^2\right\}dt\right], \tag{19}$$

which consists of scalar products of D-dimensional vectors. Each of the vector components is initially of order 1, independently of D. Even in the best of cases, i.e. if the particle positions are samples from the true posterior distribution, each of the vector components would be of the order of the true posterior standard deviation, which is equal to $1/\sqrt{2}$ independently of D. Thus the initial magnitude of change of $\tilde{N}_{\text{eff},t}$, which is the inverse of the sum of the squared weights, is also proportional to D. From this, a rough estimate of the scaling is $T(D, N, n) \propto D^{-1}$. In Fig. 1 we numerically estimated T by using trial averages, and we show the results as a function of D for different values of N. This confirms that the scaling is close to D^{-1}.

Next, we study the dependence of $T(D, N, n)$ on N, for which we rely on a numerical investigation. The results are shown in Fig. 2 for $n = 10$ and $D = 10, 20, ..., 50$. We can see that as D increases, N has to increase exponentially in order to achieve a fixed T, or in other words $T(D, N, n) \propto \log N$.

3.1.2 The effect of resampling

If the criterion to resample the particles is $\tilde{N}_{\text{eff},t} = n_{\text{crit}}$, the rate at which resampling occurs is tied to the time-scale of weight degeneracy. The immediate implication is that with a fixed ensemble size N, resampling occurs more frequently in higher dimension, with resampling rate roughly proportional to D. After resampling, $\tilde{N}_{\text{eff},t}$ is reset to N, and since the resampled particles are located at positions where the likelihood of observations is high, the initial decay of $\tilde{N}_{\text{eff},t}$ is a bit slower than for an initialization from the prior. However, since resampling does not add any new information about the true state, it cannot lead to an immediate performance increase.

The benefit of resampling is that particles with vanishing weights are discarded, and all computational efforts are expended for particles that are in an interesting region of state space. It is therefore expected that resampling shows a delayed effect due to the diffusion of particles away from resampled positions. For example, in the extreme case where $n_{\text{crit}} \approx 1$, all particles will typically be resampled at the same location. The particles have to diffuse away from their initial position such that their empirical variance is of the same order of magnitude as the (true) posterior variance. The time that is needed to reach such a state is related to the inverse of the squared diffusion coefficient and therefore independent of dimension. In Fig. 3 we show that the time-scale τ_{MSE} at which the MSE decays just after resampling, measured as the inverse absolute value of the slope of the initial decay, as a function of dimension D. The time-scale τ_{MSE} decreases with D, but it tends to

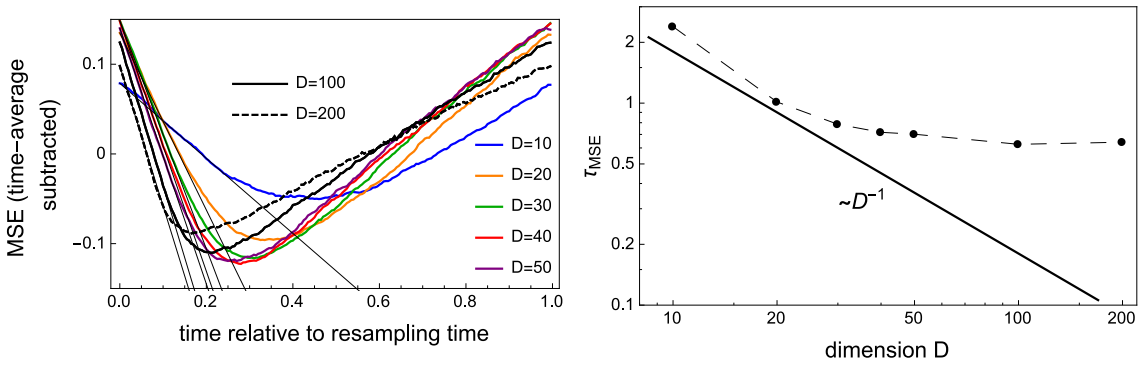

Figure 3: The effect of resampling for an ensemble size of $N = 10^4$ and a constant resampling interval of one time unit. **Left:** Resampling temporarily improves filter performance relative to the mean performance as shown by the dip in the time-course of MSE (with its time-average subtracted) after resampling. The thin straight lines show the slope of the initial decay. **Right:** The characteristic time τ_{MSE} of the MSE decrease (inverse absolute slopes of the linear functions in the left panel) decays and stabilizes, whereas the time-scale of weight degeneracy continues to decay with D^{-1} (thick black line, see also Fig. 1).

a constant value for very high D. This is in sharp contrast to the time-scale of weight degeneracy, which continues to decay with D^{-1}. Resampling is therefore ineffective in high dimensions, and it does not remove the need for exponentially large ensembles.

3.2 Optimal proposals in continuous time

In the literature on discrete time particle filters, optimal proposals for the particle motion have been shown to greatly reduce the required ensemble size. However, proposals for discrete-time filters are not directly applicable to the continuous-time case. For example, it is straightforward to show (c.f. Appendix A) that the optimal particle filter by Doucet et al. (2000) collapses to a bootstrap particle filter as the time discretization step goes to zero. Since the re-weighting of samples in the continuous-time particle filter depends on the mutual absolute continuity of the hidden state process and the particle process, the class of admissible particle motions is restricted. Two diffusion processes are mutually absolutely continuous if and only if they differ by a pure drift term. The SDE for the particle motion can therefore differ from the hidden state SDE by at most a drift term F_t:

$$dZ_t^{(i)} = \left[f(Z_t^{(i)}) + F_t^{(i)} \right] dt + g(Z_t^{(i)}) dB_t^{(i)}, \tag{20}$$

where $F_t^{(i)}$ must be a $\mathcal{F}_t^Y \vee \mathcal{F}_t^{Z^{(i)}}$-adapted D-dimensional process. The corresponding (unnormalized) weight will evolve as

$$dM_t^{(i)} = M_t^{(i)} \left(h(Z_t^{(i)}) \cdot dY_t - F_t^{(i)} \cdot dZ_t^{(i)} \right). \tag{21}$$

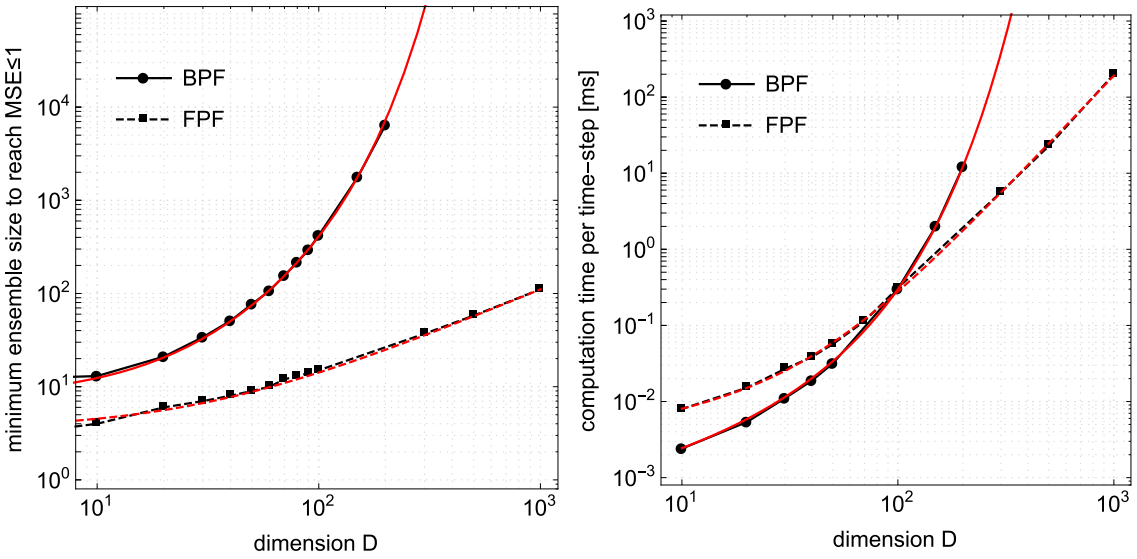

Figure 4: **Left:** Comparison of the ensemble size $N_1(D)$ that is required in order to reach a performance of MSE ≤ 1 as a function of dimension D for the Bootstrap Particle Filter (BPF) and the Feedback Particle Filter (FPF). The ensemble sizes are generally lower for the FPF and scale linearly with dimension, whereas the BPF requires an exponentially large ensemble. Fits are shown in red: $N_1^{\mathrm{BPF}}(D) \approx 31.6 \cdot e^{0.0273D} - 0.48 \cdot D - 24.3$ and $N_1^{\mathrm{FPF}}(D) \approx 3.46 + 0.107 \cdot D$. **Right:** Here, we compare the run times of the two filtering algorithms.

From this starting point, it could be interesting to formulate a stochastic control problem in order to choose the processes $F_t^{(i)}$ such as to minimize the effects of weight degeneracy. We are not aware of any work on this question and believe it to be an open problem.

It is also interesting to note that both the optimal proposal and the FPF have the aim of redirecting the particles based on the observations. However, the particle motion of an importance sampling-based particle filter is more heavily constrained in order for the importance weights to exist. Even the general form of Eq. (20) is less general than the motion of the particles in the FPF, which include an explicit dY_t term. It remains an open problem to reconcile the two frameworks of particle filtering.

3.3 Dimensionality-dependent scaling of the BPF vs. FPF

We study the minimum ensemble size that is required in order to reach a certain performance (measured as mean squared error)

$$N_\epsilon(D) = \inf \{N \in \mathbb{N} | \mathrm{MSE} \leq \epsilon\}, \tag{22}$$

where MSE is the expected time-averaged mean-squared error defined as

$$\text{MSE} = \frac{1}{Dt_1}\mathbb{E}\int_0^{t_1} dt \begin{cases} \left\| X_t - \sum_{i=1}^N m_t^{(i)} Z_t^{(i)} \right\|^2 & \text{for the BPF} \\ \left\| X_t - \frac{1}{N}\sum_{i=1}^N Z_t^{(i)} \right\|^2 & \text{for the FPF.} \end{cases} \tag{23}$$

In simulations, this integral is estimated using a Riemann sum over discrete time-steps, t_1 is chosen to be 5000 time units, for which we can drop the expectation because of the ergodicity of the process. For our particular linear toy model, the theoretical range of sensible values of ϵ is $0.5 < \epsilon < 2$, and we can set $N_{0.5}(D) = \infty$ and $N_2(D) = 1$ because the performance of the optimal filter is set to produce an MSE of 0.5 and both filters go back to the prior for $N = 1$, which yields an MSE of 2. However, the practical range of ϵ is severely restricted by the runtime of the simulations, especially for the BPF. We therefore present results for $\epsilon = 1$, which is a very low bar for a filter (not running a filter would yield the same performance), but still allows us to perform simulations to reasonably high dimensions.

The results are displayed in Fig. 4 left. Already in 10 dimensions, the FPF starts out with a significant advantage, requiring only four particles vs. 13 for the BPF. For larger dimensions, the ensemble size increases rapidly, reaching a value of 421 by D=100 and 6434 by D=200. Simulations for $D > 200$ were too time-consuming to run. Meanwhile, the FPF requires only 15 particles for $D = 100$ and an estimated 25 particles for $D = 200$. We ran simulations of the FPF up to $D = 1000$, where it requires merely 111 particles.

A least-squares fit of the numerical data reveals an exponential scaling for the BPF and merely linear scaling for the FPF (see Fig. 4 caption). The FPF thus requires roughly one additional particle with every increase of the dimension by ten. In contrast, for the same increase in dimension, the BPF requires a factor of 1.3 more particles.

In terms of computation time for a fixed performance of $MSE = 1$, we find that the BPF requires a run time per time-step that scales exponentially with dimension (see Fig. 4 right). In contrast, the FPF shows a cubic scaling, which is expected because the gain function scales quadratically with the ensemble size and the latter scales linearly with dimension. Interestingly, the FPF requires more computation time for low dimensions, despite using fewer particles to achieve the same performance as the BPF. However, we did not heavily optimize our code for performance, and we expect that using more careful programming the runtime could be reduced to compete with the BPF.

4 Discussion

In this paper, we revisited the problem of curse of dimensionality (COD) in the standard particle filter. We considered the case of the classical filtering problem with a continuous time index. Even though the COD has been studied before, all the existing literature considers only one Bayesian update step and implies a discrete-time treatment. Here, we closed this gap by studying the full dynamic nature of the problem in continuous time.

The discrete- and continuous-time particle filters have some important differences. The class of possible proposal distributions is larger in discrete time, where it is only

restricted in terms of practicality by virtue of tractability of the transition kernel. In discrete time, it has been shown that even the optimal proposal distribution does not avoid the COD (Snyder et al., 2015). In continuous time, the law of the particle motion has to be absolutely continuous with respect to the law of the hidden state. It is an open problem to show that there are nontrivial proposals that minimize the weight degeneracy.

There has been a general consensus that the problems of particle filters in high-dimensional problems result from importance sampling. It has therefore been conjectured that a particle filter without importance weights could work efficiently in high dimensions. Such a filter, the Feedback Particle Filter (FPF), has recently been proposed by Yang et al. (2013). There have other related approaches to filtering with unweighted particles, e.g. in Crişan and Xiong (2010) and Crişan and Rozovskii (2011), Chapter 23, the Ensemble Kalman Filter (Evensen, 1994; Bergemann and Reich, 2012) and the Neural Particle Filter in Kutschireiter et al. (2015). Both the Ensemble Kalman Filter and the Neural Particle Filter are mathematically similar to the FPF, and in the linear case they differ only in terms of the structure of the innovation term. Despite the promise of unweighted particle filters, their efficiency in high dimensions has not been thoroughly demonstrated so far (although there is a hint in Yang et al. (2016), the discussion seems incomplete). Here, we filled this gap and showed that the FPF avoids the COD by requiring only a polynomial ensemble size and computation time as a function of dimensionality.

Based on this important result, we want to draw the attention of researchers to the FPF and similar unweighted approaches. Particle filters without importance weights are promising algorithms for solving very high-dimensional problems. This opens up new perspectives in applied fields such as geophysics and meterology, especially numerical weather prediction, where data assimilation typically requires the solution of very high-dimensional filtering problems.

A Continuum limit of the optimal proposal from Doucet et al. (2000)

In order to apply the optimal importance function that is given in Doucet et al. (2000), Example 3, we have to introduce a time-discretized version of our diffusion process:

$$X_k = X_{k-1} + f_{\text{cont}}(X_{k-1})dt + g(X_{k-1})\sqrt{dt}\epsilon_k, \tag{24}$$

$$Y_k - Y_{k-1} = h(X_k)dt + \sqrt{dt}\eta_k, \tag{25}$$

where ϵ_k and η_k are multivariate Gaussian random variable with mean zero and unit covariance matrix. For constant $g(x) = G$ and linear $h(x) = Wx$, this corresponds to Eqs. (9,10) in Doucet et al. (2000) with $n_x = n_y = D$, $f(x) = x + f_{\text{cont}}(x)dt$, $\Sigma_v = dtGG^\top$, $C = Wdt$, and $\Sigma_w = dt\mathbb{1}_{D\times D}$, where the increment $Y_k - Y_{k-1}$ was identified with y_k of Doucet et al. (2000) in order to preserve the non-regressive form of y. Taking Eqs. (11,12) in Doucet et al. (2000) and expressing the mean m_k and covariance matrix Σ of the conditional distribution $x_k|(x_{k-1}, y_k)$ in terms of the quantities of the continuous-

time model, we obtain

$$m_k = \Sigma \left(dt^{-1} G^{-\top} G^{-1} x_{k-1} + G^{-\top} G^{-1} f_{\text{cont}}(x_{k-1}) + W^\top y_k \right), \tag{26}$$

$$\Sigma^{-1} = dt^{-1} G^{-\top} G^{-1} + dt W^\top W. \tag{27}$$

For small dt, the covariance matrix can be approximated as $\Sigma \approx dt GG^\top$, and therefore we obtain

$$m_k = x_{k-1} + f_{\text{cont}}(x_{k-1}) dt + dt GG^\top W^\top y_k. \tag{28}$$

The last term, when expressed in terms of the continuous-time process,

$$dZ_t^{(i)} = f_{\text{cont}}(Z_t^{(i)}) dt + G dW_t + GG^\top W^\top dY_t dt, \tag{29}$$

constitutes a $dt dY_t$ term, which vanishes. Therefore, the particle SDE reduces to

$$dZ_t^{(i)} = f_{\text{cont}}(Z_t^{(i)}) dt + G dW_t, \tag{30}$$

which is equal to the prior.

References

Bain, A. and D. Crişan (2009). *Fundamentals of stochastic filtering*, Volume 60 of *Stochastic Modelling and Applied Probability*. Springer, New York.

Bengtsson, T., P. Bickel, and B. Li (2008). Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems. In *Probability and Statistics: Essays in Honor of David A. Freedman*, pp. 316–334. Beachwood, Ohio, USA: Institute of Mathematical Statistics.

Bergemann, K. and S. Reich (2012). An ensemble Kalman-Bucy filter for continuous data assimilation. *Meteorologische Zeitschrift*.

Bickel, P., B. Li, and T. Bengtsson (2008). Sharp failure rates for the bootstrap particle filter in high dimensions. In *Pushing the Limits of Contemporary Statistics: Contributions in Honor of Jayanta K. Ghosh*, pp. 318–329. Beachwood, Ohio, USA: Institute of Mathematical Statistics.

Chopin, N. (2004, December). Central limit theorem for sequential Monte Carlo methods and its application to Bayesian inference. *The Annals of Statistics 32*(6), 2385–2411.

Crişan, D. (2006, October). Particle Approximations for a Class of Stochastic Partial Differential Equations. *Applied Mathematics and Optimization 54*(3), 293–314.

Crişan, D. and A. Doucet (2002). A survey of convergence results on particle filtering methods for practitioners. *IEEE Transactions on Signal Processing 50*(3), 736–746.

Crişan, D. and M. Grunwald (1998). *Large deviation comparison of branching algorithms versus resampling algorithms*. Statist. lab. Cambridge University.

Crişan, D. and B. Rozovskii (2011, February). *The Oxford Handbook of Nonlinear Filtering*. Oxford University Press.

Crişan, D. and J. Xiong (2010, February). Approximate McKean–Vlasov representations for a class of SPDEs. *Stochastics An International Journal of Probability and Stochastic Processes 82*(1), 53–68.

Daum, F. E. and J. Huang (2003). *Curse of dimensionality and particle filters*. Aerospace Conference.

Douc, R., O. Cappé, and E. Moulines (2005, October). Comparison of resampling schemes for particle filtering. In *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005*, pp. 64–69.

Doucet, A., S. Godsill, and C. Andrieu (2000, July). On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing 10*(3), 197–208.

Doucet, A. and A. M. Johansen (2009). A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering 12*, 656–704.

Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*.

Gordon, N. J., D. J. Salmond, and A. F. Smith (1993, April). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *Iee Proceedings F (Radar and Signal Processing) 140*(2), 107–113.

Künsch, H. R. (2013, September). Particle filters. *Bernoulli 19*(4), 1391–1403.

Kutschireiter, A., S. C. Surace, H. Sprekeler, and J.-P. Pfister (2015, August). A Neural Implementation for Nonlinear Filtering. *arXiv.org*.

Martino, L., V. Elvira, and F. Louzada (2017, February). Effective sample size for importance sampling based on discrepancy measures. *Signal Processing 131*(C), 386–401.

Snyder, C., T. Bengtsson, P. Bickel, and J. Anderson (2008, December). Obstacles to High-Dimensional Particle Filtering. *Monthly Weather Review 136*(12), 4629–4640.

Snyder, C., T. Bengtsson, and M. Morzfeld (2015, November). Performance Bounds for Particle Filters Using the Optimal Proposal. *Monthly Weather Review 143*(11), 4750–4761.

van Leeuwen, P. J. (2009, December). Particle Filtering in Geophysical Systems. *Monthly Weather Review 137*(12), 4089–4114.

van Leeuwen, P. J. (2010, December). Nonlinear data assimilation in geosciences: an extremely efficient particle filter. *Quarterly Journal of the Royal Meteorological Society 136*(653), 1991–1999.

Yang, T., R. S. Laugesen, P. G. Mehta, and S. P. Meyn (2016). Multivariable feedback particle filter. *Automatica*.

Yang, T., P. G. Mehta, and S. P. Meyn (2013, October). Feedback Particle Filter. *IEEE Transactions on Automatic Control 58*(10), 2465–2480.