An invitation to sequential Monte Carlo samplers: supplementary materials

S1 Particle filters and SMC samplers

General SMC method. To elucidate the connections between particle filters and SMCS, we show here that these two algorithms are specific cases of a general SMC method that approximates a sequence of target distributions $(\tilde{\pi}_t)$. Each target distribution

$$\tilde{\pi}_t(dx_{0:t}) = \tilde{\gamma}_t(x_{0:t})dx_{0:t}/\tilde{Z}_t \tag{S1.1}$$

is defined on the product space $(\mathsf{X}^{t+1}, \mathscr{X}^{t+1})$, where $\tilde{\gamma}_t(x_{0:t})$ is an unnormalized density and $\tilde{Z}_t = \int_{\mathsf{X}^{t+1}} \tilde{\gamma}_t(x_{0:t})dx_{0:t}$ is a normalizing constant (with $\tilde{Z}_0 = 1$).

The general SMC method described in Algorithm S1 combines sequential importance sampling and resampling. As input, it requires a sequence of proposal kernels (q_t) on $(\mathsf{X}, \mathscr{X})$. At step t, this defines the proposal distribution

$$\tilde{q}_t(dx_{0:t}) = \tilde{\pi}_0(dx_0) \prod_{s=1}^{t} q_s(x_{s-1}, dx_s). \tag{S1.2}$$

The weight function can be written as

$$\tilde{w}_t(x_{0:t}) = \tilde{\gamma}_t(x_{0:t})/\tilde{q}_t(x_{0:t}) = \prod_{s=1}^{t} w_s(x_{0:s}), \tag{S1.3}$$

where the incremental weight function is

$$w_t(x_{0:t}) = \frac{\tilde{\gamma}_t(x_{0:t})}{\tilde{\gamma}_{t-1}(x_{0:t-1})q_t(x_{t-1}, x_t)}. \tag{S1.4}$$

Representing the target distribution as

$$\tilde{\pi}_t(dx_{0:t}) = \prod_{s=1}^{t} w_s(x_{0:s})\tilde{q}_t(dx_{0:t})/\tilde{Z}_t \tag{S1.5}$$

allows us to adopt the Feynman–Kac formalism introduced by Del Moral (2004, 2013), where (w_t) are seen as potential functions that reassign the probability mass of $\tilde{q}_t$ to obtain

$\tilde{\pi}_t$, and the marginal distributions $\tilde{\pi}_t(dx_t) = \int_{\mathsf{X}^t} \tilde{\pi}_t(dx_{0:t})$ are referred to as (updated) Feynman–Kac models. As output, the algorithm returns weighted particles $(w_t^n, x_{0:t}^n)_{n\in[N]}$ approximating $\tilde{\pi}_t$ as $N \to \infty$, and an unbiased estimator $\tilde{Z}_t^N$ of $\tilde{Z}_t$ which is consistent as $N \to \infty$.

Algorithm S1 General sequential Monte Carlo method

```
Input: sequence of distributions $(\tilde{\pi}_t)$, proposal Markov kernels $(q_t)$, resampling
distribution $r(\cdot|w^{1:N})$ on $[N]^N$ where $w^{1:N}$ is an $N$-vector of probabilities.

  1. Initialization.
     (a) Sample particle $x_0^n$ from $\tilde{\pi}_0(\cdot)$ for $n \in [N]$ independently.
     (b) Set $w_0^n = N^{-1}$ for $n \in [N]$.
  2. For $t \in [T]$, iterate the following steps.
     (a) Sample ancestor indices $(a_{t-1}^n)_{n\in[N]}$ from $r(\cdot|w_{t-1}^{1:N})$,
         and define $\check{x}_{0:t-1}^n = x_{0:t-1}^{a_{t-1}^n}$ for $n \in [N]$.
     (b) Sample particle $x_t^n \sim q_t(\check{x}_{t-1}^n, \cdot)$ and set $x_{0:t}^n = (\check{x}_{0:t-1}^n, x_t^n)$ for $n \in [N]$.
     (c) Compute weights $w_t(x_{0:t}^n)$ for $n \in [N]$, and set $w_t^n \propto w_t(x_{0:t}^n)$ such that
         $\sum_{n\in[N]} w_t^n = 1$.

Output: weighted particles $(w_t^n, x_{0:t}^n)_{n\in[N]}$ approximating $\tilde{\pi}_t$, and estimator
$\tilde{Z}_t^N = \prod_{s=1}^t N^{-1} \sum_{n\in[N]} w_s(x_{0:s}^n)$ of $\tilde{Z}_t$ for $t \in [T]$.
```

Particle filters. We now introduce state space models which are also known as hidden Markov models. Consider a latent Markov chain $(x_t)_{t\geq 0}$ defined on $(\mathsf{X}, \mathscr{X})$, initialized as $x_0 \sim \pi_0$ and evolving for each time step $t \geq 1$ according to a Markov kernel f, i.e. $x_t|x_{t-1} \sim f(x_{t-1}, \cdot)$. We assume access to Y-valued observations $(y_t)_{t\geq 1}$ that are modelled as conditionally independent given $(x_t)_{t\geq 0}$, with observation density g on $(\mathsf{Y}, \mathscr{Y})$, i.e. $y_t|x_t \sim g(x_t, \cdot)$.

Given observations collected up to time t, sequential state inference is based on the posterior distribution

$$p(dx_{0:t}|y_{1:t}) = \frac{p(dx_{0:t})p(y_{1:t}|x_{0:t})}{p(y_{1:t})}, \tag{S1.6}$$

where the joint distribution of the states is $p(dx_{0:t}) = \pi_0(dx_0)\prod_{s=1}^{t} f(x_{s-1}, dx_s)$ and the conditional likelihood of the observations is $p(y_{1:t}|x_{0:t}) = \prod_{s=1}^{t} g(x_s, y_s)$. We will also be interested in the marginal likelihood $p(y_{1:t}) = \int_{\mathsf{X}^{t+1}} p(dx_{0:t}, y_{1:t})$ when there are unknown parameters in the model to be inferred. From (S1.6), we can derive other quantities of interest such as the filtering distribution $p(dx_t|y_{1:t})$, defined as the last marginal of $p(dx_{0:t}|y_{1:t})$, and the state predictive distribution $p(dx_{t+1}|y_{1:t}) = \int_{\mathsf{X}} f(x_t, dx_{t+1})p(dx_t|y_{1:t})$. Particle filters can be understood as specific cases of SMC methods to sequentially approximate the posterior distribution $\tilde{\pi}_t(dx_{0:t}) = p(dx_{0:t}|y_{1:t})$ (with $\tilde{\pi}_0 = \pi_0$) and the marginal likelihood $\tilde{Z}_t = p(y_{1:t})$. In this setting, the incremental weight function in (S1.4) reduces to

$$w_t(x_{t-1}, x_t) = \frac{f(x_{t-1}, x_t)g(x_t, y_t)}{q_t(x_{t-1}, x_t)}. \tag{S1.7}$$

Different choices of proposal kernels (q_t) give rise to distinct SMC methods. For example, the bootstrap particle filter of Gordon et al. (1993) corresponds to Algorithm S1 with $q_t(x_{t-1}, dx_t) = f(x_{t-1}, dx_t)$ and $w_t(x_t) = g(x_t, y_t)$ for all t.

SMC samplers. We now cast SMCS presented in this article as specific cases of SMC methods. Given a sequence of target distributions (π_t) and backward Markov kernels (L_t) on $(\mathsf{X}, \mathscr{X})$, the target distribution in (S1.1) is

$$\tilde{\pi}_t(dx_{0:t}) = \pi_t(dx_t) \prod_{s=1}^{t} L_{s-1}(x_s, dx_{s-1}), \tag{S1.8}$$

with $\tilde{\pi}_0 = \pi_0$. Note that (S1.8) has π_t as the marginal distribution on x_t and the normalizing constant is $\tilde{Z}_t = Z_t$. In this case, the proposal kernels (q_t) correspond to the forward kernels (M_t) defined on $(\mathsf{X}, \mathscr{X})$ and the incremental weight function (S1.4) reduces to the weight function in (2.1).

Under these settings, one can check that Algorithm S1 recovers the generic SMCS described in Algorithm 1, with the exception that we only keep track of the particles approximating the target distribution at each step. The latter is sufficient due to the simplified form of the weight function (2.1) and the fact that only the terminal time marginal distribution of (S1.8) is of interest.

In summary, we see that particle filters and SMCS are instances of SMC methods that approximate different sequences of target distributions $(\tilde{\pi}_t)$, defined by either the specificity of the problem in (S1.6) or algorithmic choices in (S1.8) via the specification of backward kernels. The terminal time marginal distribution $\tilde{\pi}_t(dx_t)$ and normalizing constant $\tilde{Z}_t$ represent the filtering distribution $p(dx_t|y_{1:t})$ and marginal likelihood $p(y_{1:t})$ in particle filtering, and a target distribution $\pi_t(dx_t)$ and its associated normalizing constant Z_t for SMCS. The proposal kernel q_t corresponds to the state transition f in the case of a bootstrap particle filter and the forward Markov kernel M_t in SMCS.

S2 Unadjusted Hamiltonian Monte Carlo moves

As an alternative to the unadjusted Langevin moves described in Section 2.3, we can consider kernels constructed using Hamiltonian dynamics (Duane et al. 1987) that target $\tilde{\pi}_t(dx_t, dv_t) = \pi_t(dx_t)\mathcal{N}(dv_t; 0, \Omega)$ for $(x_t, v_t) \in \mathbb{R}^d \times \mathbb{R}^d$. Here x_t are the original states, v_t are auxiliary variables and $\Omega \in \mathbb{R}^{d\times d}$ denotes a "mass matrix". Given a sample x_{t-1} from π_{t-1} at step $t-1$, we sample $v_{t-1} \sim \mathcal{N}(0, \Omega)$, so that the pair (x_{t-1}, v_{t-1}) follows $\tilde{\pi}_{t-1}$. We define the initial position $q(0) = x_{t-1}$ and initial momentum $p(0) = v_{t-1}$ of a fictitious object undergoing Hamiltonian dynamics, with Hamiltonian function $H_t(q, p) = -\log \pi_t(q) + p^\top \Omega^{-1} p/2$. The associated dynamics is commonly discretized using the leap-frog integrator, with a step size $\varepsilon > 0$ and a number of steps $m \in \mathbb{N}$, yielding a trajectory $(q(\ell), p(\ell))$ for $\ell = 1, \dots, m$. Finally, we set $x_t = q(m)$ and $v_t = p(m)$. We write the composition of leap-frog iterations as $\Phi_t^\ell(q(0), p(0)) = (q(\ell), p(\ell))$ for $\ell \in [m]$. The transition from (x_{t-1}, v_{t-1}) to (x_t, v_t) defines a deterministic forward kernel M_t, namely a Dirac mass on $\Phi_t^m(x_{t-1}, v_{t-1})$.

As the Hamiltonian is not conserved exactly under time-discretization, M_t is not $\tilde{\pi}_t$-invariant. It is again possible to correct the discretization error using importance sampling to target $\tilde{\pi}_t(dx_t, dv_t)$ with proposal $q_t(dx_t, dv_t) = (\tilde{\pi}_{t-1} \# \Phi_t^m)(dx_t, dv_t)$. The # notation refers to the push-forward operator, so that $(\tilde{\pi}_{t-1} \# \Phi_t^m)$ is the measure obtained by sampling from $\tilde{\pi}_{t-1}$ and applying the map Φ_t^m. Using reversibility and volume preservation properties of Φ_t^m, the proposal density can be computed using change of variables, i.e. $q_t(x_t, v_t) = \tilde{\pi}_{t-1}(x_{t-1}, v_{t-1})$ where $(x_{t-1}, v_{t-1}) = (\Phi_t^m)^{-1}(x_t, v_t)$ is obtained using the inverse map. The resulting importance weight is

$$w_t(x_{t-1}, v_{t-1}, x_t, v_t) \propto \frac{\tilde{\pi}_t(x_t, v_t)}{\tilde{\pi}_{t-1}(x_{t-1}, v_{t-1})} = \frac{\exp(-H_t(x_t, v_t))}{\exp(-H_{t-1}(x_{t-1}, v_{t-1}))}, \tag{S2.1}$$

which corresponds to the choice of backward kernel

$$L_{t-1}((x_t, v_t), dx_{t-1}, dv_{t-1}) = \delta_{(\Phi_t^m)^{-1}(x_t, v_t)}(dx_{t-1}, dv_{t-1}). \tag{S2.2}$$

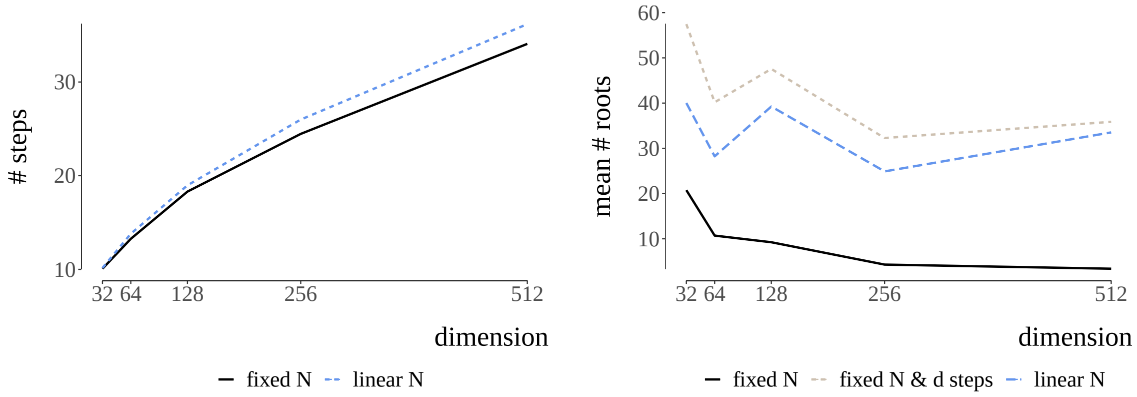
The above arguments and related ideas can be found in Jarzynski (2000), Neal (2005), Schöll-Paschinger & Dellago (2006), and have been recently employed in variational inference frameworks to tune algorithmic parameters (Geffner & Domke 2021, Zhang et al. 2021).

We now discuss some extensions of the above framework. Firstly, one could replace Φ_t^m with other reversible and volume-preserving maps. Secondly, we can consider several iterations of momentum refreshment and leap-frog integration, i.e. initializing at $x_{t,0} = x_{t-1}$, we would sample $\tilde{v}_{t,i-1} \sim \mathcal{N}(0, \Omega)$ and set $(x_{t,i}, v_{t,i}) = \Phi_t^m(x_{t,i-1}, \tilde{v}_{t,i-1})$ for $i \in [I]$. One could also benefit from partial momentum refreshment (Horowitz 1991) by updating $\tilde{v}_{t,i}$ with an autoregressive process that leaves $\mathcal{N}(0, \Omega)$ invariant. In contrast to compositions of π_t-invariant kernels that do not affect importance weights, we have to modify (S2.1) to account for the additional iterations. Thirdly, in the spirit of the work by Neal (1994), Nishimura & Dunson (2018) for HMC and Dau & Chopin (2022) for SMC, we can use all iterates in the leap-frog integrator instead of only the terminal ones, by considering all the proposals $q_t^\ell(dx_t, dv_t) = (\tilde{\pi}_{t-1} \# \Phi_t^\ell)(dx_t, dv_t)$ for all $\ell \in [m]$ when forming an importance sampling approximation of $\tilde{\pi}_t(dx_t, dv_t)$. In Algorithm 1, one would have $N \times m$ instead of N samples to consider in Steps 2(b) and 2(c); the resampling operation in Step 2(a) would then select N particles among the $N \times m$ weighted samples. Since the use of multiple proposals within importance sampling is consistent in the limit of the number of samples, it follows that the resulting SMCS will also be consistent as $N \to \infty$.

S3 Numerical experiments on Gaussians

We consider numerical experiments on Normal distributions in varying dimensions d. We set $\pi_0(dx) = \mathcal{N}(x; \mu_0, \Sigma_0)dx$ with $\mu_0 = (1, \dots, 1)$, $\Sigma_0 = \text{diag}(0.5, \dots, 0.5)$ and $\pi(dx) = \mathcal{N}(x; \mu, \Sigma)dx$ with $\mu = (0, \dots, 0)$, $\Sigma = \text{diag}(1, \dots, 1)$. Despite the simple setup, standard importance sampling would give rise to estimators with infinite variance. We consider a geometric path of distributions, all of which are Normal. For each $t \in [T]$, we employ a π_t-invariant HMC kernel for M_t, with step size $\varepsilon = d^{-1/4}$ and $m = \lceil d^{1/4} \rceil$ leap-frog steps. The "mass matrix" Ω is taken as diagonal and adapted using the empirical marginal precisions computed from the particle approximations. The backward kernel L_{t-1} is taken as the time reversal of M_t. All simulations employ multinomial resampling.

Figure S1a shows the number of bridging distributions T obtained using the adaptive strategy in Section 2.4, with an ESS threshold of $\kappa = 0.5$. The two lines correspond to having $N = 256$ ("fixed N") and $N = 256 + 8d$ ("linear N") number of particles. The resulting T appears to increase sub-linearly with d in both regimes. We introduce a setup referred to as "fixed N & d steps" in the plots, where $N = 256$ and $T = d$; thus inserting more intermediate distributions than required for controlling the ESS. In this setup, $(\lambda_t)_{t\in[T]}$ was determined by interpolating between the inverse temperatures obtained from an adaptive SMCS run. The interpolation was performed using the `cobs` package in R (Ng & Maechler 2007), which allows one to fit splines constrained to be monotonically increasing. In the "fixed N & d steps" setup, once the sequence $(\lambda_t)_{t\in[T]}$ is obtained, we run SMCS with adaptive tuning of the MCMC kernels, we store all quantities required to define these kernels and we re-run SMCS with these kernels fixed. That last run generates unbiased estimators of $\hat{Z}$. Implementation details can be followed in the accompanying R scripts.



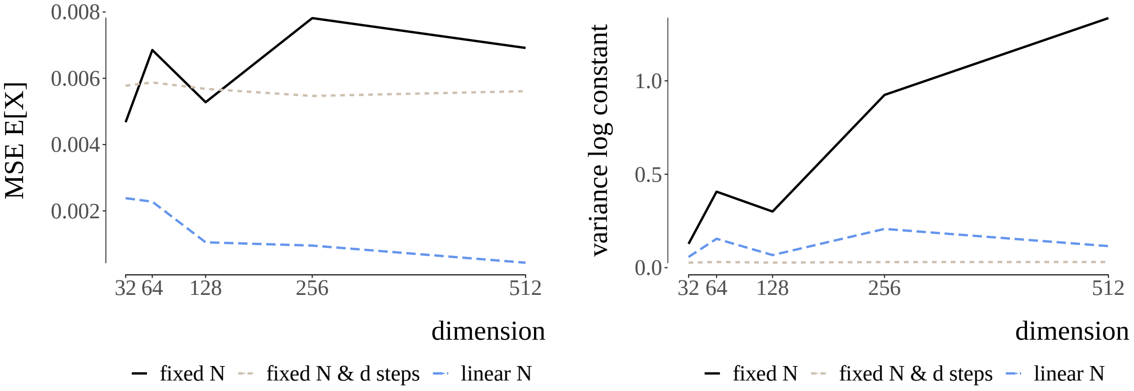
(a) Number of bridging distributions.

(b) Number of roots.

Figure S1: Normal example of Section S3. Number of distributions T chosen by an adaptive SMCS (*left*). Number of roots in the genealogical trees, in different regimes (*right*). Plots are obtained by averaging 100 independent repeats.

To compare the three setups and also illustrate some of the discussion in Section 4 of the article, we consider the number of "roots" or unique ancestors in the genealogical trees of the particle systems and plot its average over repeated runs against dimension in Figure S1b. We observe that the number of roots at the terminal time decreases in the "fixed N" regime. However, it appears stable when either N increases linearly with d in the adaptive SMCS, or when T scales linearly with d for fixed N. This suggests that the sampler is stable with d in these two regimes, where either the number of particles or the number of intermediate steps increases adequately.

We investigate these regimes further using common measures of Monte Carlo error in Figure S2, such as the mean squared error (MSE) associated with the SMCS estimator of $\mathbb{E}_\pi[X] = \int_{\mathsf{X}} x\pi(dx)$ (averaged over all components in Figure S2a). The MSE appears stable in all regimes and even decreases when N increases with d. Figure S2b displays



(a) MSE for estimator of $\mathbb{E}_\pi[X]$.

(b) Variance of $\log Z_T^N$.

Figure S2: Normal example of Section S3. MSE for estimator of $\mathbb{E}_\pi[X]$ (*left*) and variance of $\log Z_T^N$ (*right*) with increasing dimension. Plots are obtained by averaging 100 independent repeats.

the variance of the logarithm of the normalizing constant estimator Z_T^N against dimension, which appears to increase with d for “fixed N”, but looks stable in the other two regimes, matching the behaviour of the number of roots in Figure S1b. These results confirm that SMCS can indeed deliver a stable accuracy for a polynomial cost as the dimension d increases, in settings where importance sampling would fail.

S4 Details concerning the experiments of the article

We provide the implementation details of SMCS that was used to generate the figures in Section 5 of the article. The problem settings are summarized in Table S1.

To obtain Figures 2a-2b we proceeded as follows. We assimilate the data in batches of size 10, and we introduce intermediate distributions using a geometric path between successive partial posteriors $p(d\beta|x_{1:10t}, y_{1:10t})$ and $p(d\beta|x_{1:10(t+1)}, y_{1:10(t+1)})$, with the convention that $p(d\beta|x_{1:10t}, y_{1:10t})$ is $p(d\beta)$ (the prior distribution) for $t = 0$. Each SMCS run employs $N = 1024$ particles, an ESS threshold of $\kappa = 0.5$ to determine intermediate distributions between two partial posteriors, and 2 HMC iterations per move step with $\varepsilon = 0.3 \times d^{-1/4}$ and $\lceil \varepsilon^{-1} \rceil$ leap-frog steps. Backward kernels are taken as time reversals of the forward kernels, and multinomial resampling was employed.

To obtain Figures 2c-2d we proceeded similarly but with batches of data of size 100. Tuning choices was otherwise identical to the above description.

Moving on to Section 5.2, we employed Stan (Carpenter et al. 2017) to obtain evaluations of the target density and its gradient. The function `integrate_ode_rk45` of Stan was used to solve the SIR ordinary differential equation. To obtain the partial posteriors shown in Figures 3c-3d we first ran a basic MCMC algorithm targeting the posterior distribution given the first three observations. Specifically we employed a Metropolis–Rosenbluth–Teller–Hastings algorithm with Normal random walk proposals, using a covariance matrix adapted during initial runs, and using a diagonal matrix with entries 0.01 in the very first run, along with the starting state $\log\gamma = -1$, $\log\beta = 1$, $\log\phi_{\text{inv}} = -1$. We then calibrated a 3-dimensional Normal distribution using the mean and variance of the MCMC samples, to define π_0. Specifically we obtained

$$\pi_0(\log\gamma, \log\beta, \log\phi_{\text{inv}}) = \mathcal{N}\left(\begin{pmatrix}-0.97\\0.49\\-2.6\end{pmatrix}, \begin{pmatrix}1.59 & 0.25 & 0.16\\0.25 & 0.06 & 0.04\\0.16 & 0.04 & 1.08\end{pmatrix}\right),$$

for the transformed parameters $(\log\gamma, \log\beta, \log\phi_{\text{inv}})$. We employed geometric paths interpolating between π_0 and the posterior distribution given three observations, and then between successive partial posteriors, assimilating observations one by one. We ran SMCS

	Section 5.1	**Section 5.2**
Data set	Forest cover type data set with covariates $x = (x_1, \ldots, x_m)$ and cover type $y = (y_1, \ldots, y_m)$	English boarding school data set with daily counts $y = (y_1, \ldots, y_m)$ of pupils confined to bed
Model	Logistic regression model $p(y \mid x, \beta) = \prod_{i=1}^{m} \mathcal{B}(y_i; (1 + \exp(-x_i^\top \beta))^{-1})$	Deterministic SIR model $(S_t, I_t, R_t)_{t \geq t_0}$ $p(y \mid t_0, \theta) = \prod_{t=1}^{m} \mathcal{NB}(y_t; I_t, \phi_{\text{inv}})$
Parameters	β contains regression coefficients	$\theta = (\gamma, \beta, \phi_{\text{inv}})$ contains infection rate, recovery rate and dispersion
Prior distribution	$p(\beta) = \prod_{i=1}^{d} \mathcal{N}(\beta_i; 0, 10)$	$p(\theta) = \mathcal{TN}(\gamma; 0.4, 0.5^2)\mathcal{TN}(\beta; 2, 1)\mathcal{E}(\phi_{\text{inv}}; 5)$
Target distribution π	Posterior distribution $\pi(\beta) = p(\beta \mid x, y)$	Posterior distribution $\pi(\theta) = p(\theta \mid t_0, y)$
Normalizing constant Z	Marginal likelihood $Z = p(y \mid x)$	Marginal likelihood $Z(t_0) = p(y \mid t_0)$

Table S1: Summary of examples in Section 5 of the article. $\mathcal{B}$ denotes the Bernoulli distribution, $\mathcal{NB}$ the Negative Binomial distribution, $\mathcal{TN}$ the Truncated Normal distribution with support on $\mathbb{R}_+$ and $\mathcal{E}$ the exponential distribution.

with $N = 512$ particles, ESS threshold of $\kappa = 0.5$, 2 HMC iterations per move step, with stepsize of $\varepsilon = 0.1$ and 10 leap-frog steps. Each backward kernel was set as the time reversal of the associated forward kernel. Multinomial resampling was employed.

To obtain the marginal likelihood plot in Figure 3b, we employed SMCS using the path of partial posteriors, with geometric paths interpolating between successive partial posteriors, and started from π_0, exactly as described above. The only difference is that we used $N = 4096$ particles and 3 HMC iterations per move step, resulting in a smaller variation across 5 independent runs.

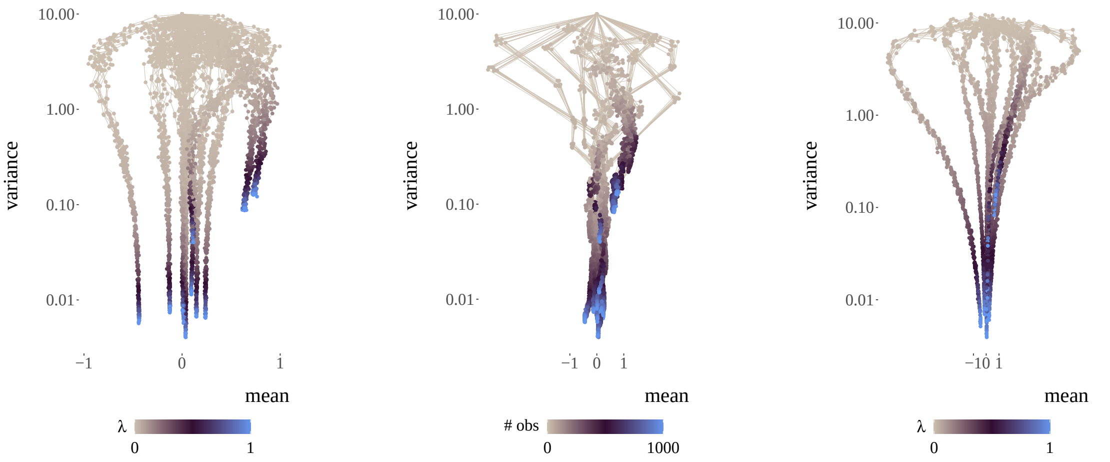
S5 Logistic regression: different paths

We consider the logistic regression example described in Section 5.1 of the article, and mentioned also in Section 2.2. We used $N = 1024$ particles and an ESS threshold of $\kappa = 0.5$. Using the first $m = 1000$ rows of the data, Figure S3 shows the mean and variance of the $d = 11$ components of β for three paths of distributions: a geometric path (S3a), a path of partial posteriors where observations are assimilated in batches of size 10 (S3b), and a path of "least coding effort" using the Pólya–Gamma Gibbs sampler (S3c) with scaled covariates $\lambda_t x$ and $\lambda_t \in [0, 1]$. HMC moves were employed for the first two paths, with stepsize $\varepsilon = 0.3 \times d^{-1/4}$, $\lceil \varepsilon^{-1} \rceil$ leap-frog steps, and 10 independent repeats are shown. In all cases, we set the backward kernels to be the time reversal of the corresponding forward kernels, and employed multinomial resampling for all simulations. The three paths start and end at the same distributions but the intermediate distributions are visibly different.

S6 Partial posterior averaged over orderings

We illustrate the usefulness of unbiased estimators in the setting of logistic regression, as in Section 5.1 of the article. As described in Middleton et al. (2019), unbiased estimators can be generically obtained from SMCS through the coupling of an MCMC scheme proposed in Andrieu et al. (2010), known as "particle independent Metropolis–Hastings" (PIMH).

Suppose that one of the covariates in the regression is actually a random draw from another model, as in two-step estimation (Murphy & Topel 2002), and that we want to propagate that uncertainty onto the posterior. In the Bayesian terminology, this could lead to a "cut distribution" (Plummer 2015). The same computational challenge occurs when some data are missing and multiple imputation is performed (Rubin 1996), or in



(a) Geometric path. (b) Partial posteriors. (c) Path of least coding.

Figure S3: Three paths of distributions connecting the prior to the posterior in a logistic regression example described in Section 5.1 of the article, represented by lines in the mean-variance plane, for each component and 10 independent runs.

the context of propensity scores (Zigler & Dominici 2014), or when bagging posteriors (Bühlmann 2014). All these cases are instances of the generic question of approximating $\pi(dx) = \int \pi(dx|\eta)g(d\eta)$, where we can sample from $\eta \sim g$ and design a Monte Carlo method to approximate the conditional distribution $\pi(dx|\eta)$. With unbiased approximations of $\pi(dx|\eta)$ for any $\eta \sim g$, we obtain an unbiased approximation of $\pi(dx)$ itself.

Here we consider a variant of the path of partial posteriors for logistic regression, $\pi_m(d\beta) = p(d\beta|x_{1:m}, y_{1:m})$ given m observations. This path usually depends on a specific ordering of the observations. As an alternative we consider the posterior distribution averaged over orderings, $\pi^\star_m(d\beta) = (m!)^{-1}\sum_\sigma p(d\beta|x_{\sigma(1:m)}, y_{\sigma(1:m)})$ where $\sigma(1:m)$ denotes the first m elements of a permutation of the entire data set. To obtain unbiased estimators, we sample m observations from the data set at random without replacement, and run coupled PIMH chains (Middleton et al. 2019) targeting $p(d\beta|x_{\sigma(1:m)}, y_{\sigma(1:m)})$ using SMCS with $N = 128$ particles as proposals. Each SMCS run employs HMC moves with stepsize $\varepsilon = 0.3 \times d^{-1/4}$ and 3 leap-frog steps. The schedule and the mass matrices are obtained using an initial run of adaptive SMCS on a data set of the same size m, and these tuning parameters are then fixed in the generation of unbiased estimators. Indeed the use of adaptive techniques would jeopardize the lack-of-bias property of Z_T^N and thus could change the target distribution of the PIMH kernels. We compute $R = 1000$ independent unbiased estimators for each m, and use empirical averages to approximate $\pi_m^\star(d\beta)$. Figure S4 illustrates the evolution of the means and variances of β as m increases. These distributions can be interpreted as the posterior distributions given m randomly chosen observations from the data set, as opposed to conditioning on the first m observations in an arbitrary order.

S7 Laplace approximations to initialize SMCS

Bayesian asymptotics provide useful strategies for Monte Carlo computation. For example, we can use a Laplace approximation of the posterior as initial distribution π_0, i.e. a Normal distribution centered at the maximum likelihood estimate (MLE) and with covariance given by the inverse of the information matrix at the MLE (e.g. Chopin & Ridgway 2017).

In the context of Section 5.1, Figure S5a shows that the approximation is extremely accurate when the number of observations m is large and leads to very high effective sample sizes in a single step of importance sampling. Here SMCS employs $N = 1024$ particles, an ESS threshold of $\kappa = 0.5$ and HMC moves, with step size $\varepsilon = 0.3 \times d^{-1/4}$ and $\lceil \varepsilon^{-1} \rceil$ leap-frog steps. The relative variance of Z_T^N is close to zero for data sizes above $m = 10^4$. Thus

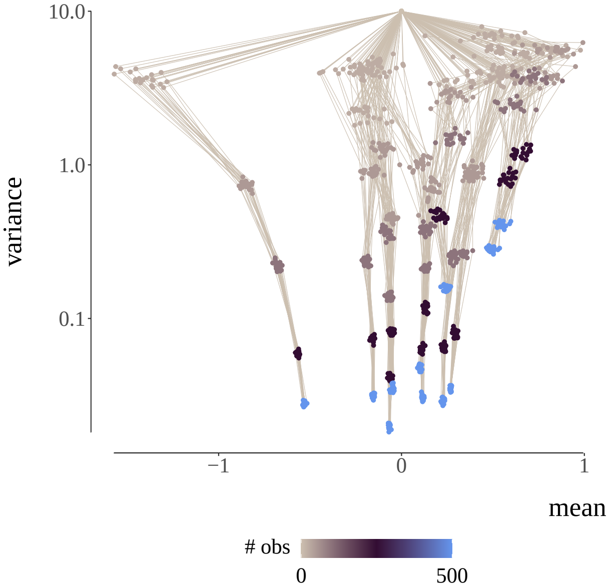
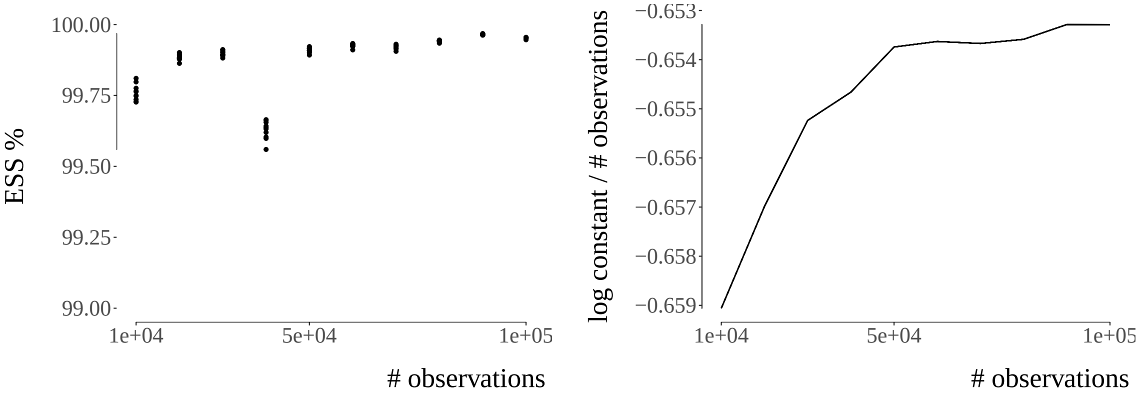


Figure S4: Logistic regression with forest cover type data. Path of partial posteriors averaged over orderings of the data. The ten realizations are obtained by non-parametric bootstrap from $R = 1000$ independent unbiased estimators.

SMCS that employ the ESS criterion to select the next inverse temperature would default back to plain importance sampling. Figure S5b shows the estimates of $\log Z$ divided by the number of observations m; ten repeats are overlaid but the accuracy is such that they are indistinguishable. Some effects of Bayesian asymptotics on the performance of SMCS are studied in Chopin (2002), samplers for tall data settings are proposed in Gunawan et al. (2020), and approximate samplers for sequential inference in Gerber & Douc (2020).



(a) Effective sample size.

(b) Estimates of $\log Z/m$.

Figure S5: Logistic regression with forest cover type data. ESS against number of observations m (*left*) and estimates of $\log Z/m$ (*right*), when initializing from a Laplace approximation of the posterior.

References

Andrieu, C., Doucet, A. & Holenstein, R. (2010), 'Particle Markov chain Monte Carlo methods', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **72**(3), 269–342. S6

Bühlmann, P. (2014), 'Discussion of Big Bayes stories and BayesBag', *Statistical Science* **29**(1), 91–94. S6

Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P. & Riddell, A. (2017), 'Stan: A probabilistic programming language', *Journal of Statistical Software* **76**(1). S4

Chopin, N. (2002), 'A sequential particle filter method for static models', *Biometrika*

89(3), 539–552. S7

Chopin, N. & Ridgway, J. (2017), 'Leave Pima Indians alone: binary regression as a benchmark for Bayesian computation', *Statistical Science* **32**(1), 64–87. S7

Dau, H.-D. & Chopin, N. (2022), 'Waste-free sequential Monte Carlo', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* . S2

Del Moral, P. (2004), *Feynman–Kac formulae*, Springer. S1

Del Moral, P. (2013), *Mean field simulation for Monte Carlo integration*, CRC press. S1

Duane, S., Kennedy, A. D., Pendleton, B. J. & Roweth, D. (1987), 'Hybrid Monte Carlo', *Physics letters B* **195**(2), 216–222. S2

Geffner, T. & Domke, J. (2021), 'MCMC variational inference via uncorrected Hamiltonian annealing', *Advances in Neural Information Processing Systems* **34**. S2

Gerber, M. & Douc, R. (2020), 'Online approximate Bayesian learning', *arXiv preprint arXiv:2007.04803* . S7

Gordon, N. J., Salmond, D. J. & Smith, A. F. M. (1993), Novel approach to nonlinear/non-Gaussian Bayesian state estimation, *in* 'IEE Proceedings F (Radar and Signal Processing)', Vol. 140(2), IET, pp. 107–113. S1

Gunawan, D., Dang, K.-D., Quiroz, M., Kohn, R. & Tran, M.-N. (2020), 'Subsampling sequential Monte Carlo for static Bayesian models', *Statistics and Computing* **30**(6), 1741–1758.

URL: *https://doi.org/10.1007/s11222-020-09969-z* S7

Horowitz, A. M. (1991), 'A generalized guided Monte Carlo algorithm', *Physics Letters B* **268**(2), 247–252. S2

Jarzynski, C. (2000), 'Hamiltonian derivation of a detailed fluctuation theorem', *Journal of Statistical Physics* **98**, 77–102. S2

Middleton, L., Deligiannidis, G., Doucet, A. & Jacob, P. E. (2019), Unbiased smoothing using particle independent Metropolis-Hastings, *in* K. Chaudhuri & M. Sugiyama, eds, 'Proceedings of Machine Learning Research', Vol. 89, PMLR, pp. 2378–2387. S6

Murphy, K. M. & Topel, R. H. (2002), 'Estimation and inference in two-step econometric models', *Journal of Business & Economic Statistics* **20**(1), 88–97. S6

Neal, R. M. (1994), 'An improved acceptance procedure for the hybrid Monte Carlo algorithm', *Journal of Computational Physics* **111**(1), 194–203. S2

Neal, R. M. (2005), Hamiltonian importance sampling, *in* 'talk presented at the Banff International Research Station (BIRS) workshop on Mathematical Issues in Molecular Dynamics'. S2

Ng, P. & Maechler, M. (2007), 'A fast and efficient implementation of qualitatively constrained quantile smoothing splines', *Statistical Modelling* **7**(4), 315–328. S3

Nishimura, A. & Dunson, D. (2018), 'Recycling intermediate steps to improve Hamiltonian Monte Carlo', *Bayesian Analysis* . S2

Plummer, M. (2015), 'Cuts in Bayesian graphical models', *Statistics and Computing* **25**(1), 37–43. S6

Rubin, D. B. (1996), 'Multiple imputation after 18+ years', *Journal of the American Statistical Association* **91**(434), 473–489. S6

Schöll-Paschinger, E. & Dellago, C. (2006), 'A proof of Jarzynski's nonequilibrium work theorem for dynamical systems that conserve the canonical distribution', *The Journal of Chemical Physics* **125**(5), 054105. S2

Zhang, G., Hsu, K., Li, J., Finn, C. & Grosse, R. B. (2021), 'Differentiable annealed importance sampling and the perils of gradient noise', *Advances in Neural Information Processing Systems* **34**. S2

Zigler, C. M. & Dominici, F. (2014), 'Uncertainty in propensity score estimation: Bayesian methods for variable selection and model-averaged causal effects', *Journal of the American Statistical Association* **109**(505), 95–107. S6