Variational Sequential Monte Carlo

Christian A. Naesseth
Linköping University

Scott W. Linderman
Columbia University

Rajesh Ranganath
New York University

David M. Blei
Columbia University

Abstract

Many recent advances in large scale probabilistic inference rely on variational methods. The success of variational approaches depends on (i) formulating a flexible parametric family of distributions, and (ii) optimizing the parameters to find the member of this family that most closely approximates the exact posterior. In this paper we present a new approximating family of distributions, the variational sequential Monte Carlo (VSMC) family, and show how to optimize it in variational inference. VSMC melds variational inference (VI) and sequential Monte Carlo (SMC), providing practitioners with flexible, accurate, and powerful Bayesian inference. The VSMC family is a variational family that can approximate the posterior arbitrarily well, while still allowing for efficient optimization of its parameters. We demonstrate its utility on state space models, stochastic volatility models for financial data, and deep Markov models of brain neural circuits.

1 Introduction

Complex data like natural images, text, and medical records require sophisticated models and algorithms. Recent advances in these challenging domains have relied upon variational inference (VI) [Kingma and Welling, 2014, Hoffman et al., 2013, Ranganath et al., 2016a]. Variational inference excels in quickly approximating the model posterior, yet these approximations are only useful insofar as they are accurate. The challenge is to balance faithful posterior approximation and fast optimization.

We present a new approximating family of distributions called variational sequential Monte Carlo (VSMC). VSMC blends VI and sequential Monte Carlo (SMC) [Stewart and McCarty, 1992, Gordon et al., 1993, Kitagawa, 1996], providing practitioners with a flexible, accurate, and powerful approximate Bayesian inference algorithm. VSMC is an efficient algorithm that can approximate the posterior arbitrarily well.

Standard SMC approximates a posterior distribution of latent variables with N weighted particles iteratively drawn from a proposal distribution. The idea behind *variational* SMC is to view the parameters of the proposal as indexing a family of distributions over latent variables. Each distribution in this variational family corresponds to a particular choice of proposal; to sample the distribution, we run SMC to generate a set of particles and then randomly select one with probability proportional to its weight. Unlike typical variational families, the VSMC family trades off fidelity to the posterior with computational complexity: its accuracy increases with the number of particles N, but so does its computational cost.

We develop the VSMC approximating family, derive its corresponding variational lower bound, and design a stochastic gradient ascent algorithm to optimize its parameters. We connect VSMC to the importance weighted auto-encoder (IWAE) [Burda et al., 2016] and show that the IWAE lower bound is a special case of the VSMC bound. As an illustration, consider approximating the following posterior with latent variables $x_{1:T}$ and observations $y_{1:T}$,

$$p(x_{1:T} \mid y_{1:T}) = \prod_{t=1}^{T} \mathcal{N}(x_t\,; 0, 1)\,\mathcal{N}(y_t\,; x_t^2, 1)/p(y_{1:T}).$$

This is a toy Gaussian state space model (SSM) where the observed value at each time step depends on the square of the latent state. Figure 1c shows the approximating power of VSMC versus that of the IWAE and of standard variational Bayes (VB). As the length of the sequence T increases, naïve importance sampling effectively collapses to use only a single particle. VSMC on the other hand maintains a diverse set of particles and thereby achieves a significantly tighter lower bound of the log-marginal likelihood $\log p(y_{1:T})$.

We focus on inference in state space and time series models, but emphasize that VSMC applies to any sequence of probabilistic models, just like standard SMC [Del Moral et al., 2006, Doucet and Johansen, 2009, Naesseth et al., 2014].

Proceedings of the 21^{st} International Conference on Artificial Intelligence and Statistics (AISTATS) 2018, Lanzarote, Spain. PMLR: Volume 84.

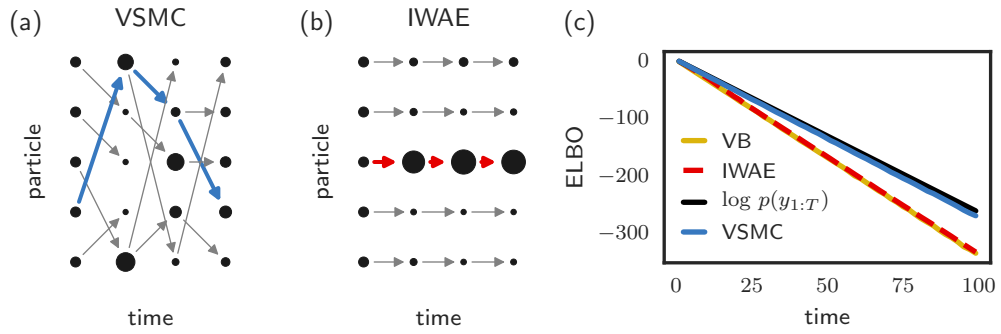


Figure 1: *Comparing* VSMC *and the* IWAE. (a) VSMC constructs a weighted set of particle trajectories using SMC and then samples one according to the final weight. Here, the size of the dot is proportional to the weight, w_t^i; the gray arrows denote the ancestors, a_{t-1}^i; and the blue arrows denote the chosen path, $b_{1:T}$. (b) IWAE does the same, but without resampling. This leads to particle degeneracy as time increases—only one particle has nonneglible weight at time T. (c) The ELBO suffers from this degeneracy: all are comparable when T is small, but as time increases the IWAE provides minimal improvement over standard VB, whereas VSMC still achieves nearly the true marginal likelihood.

In Section 5, we demonstrate the advantages of VSMC on both simulated and real data. First, we show on simulated linear Gaussian SSM data that VSMC can outperform the (locally) optimal proposal [Doucet et al., 2001, Doucet and Johansen, 2009]. Then we compare VSMC with IWAE for a stochastic volatility model on exchange rates from financial markets. We find that VSMC achieves better posterior inferences and learns more efficient proposals. Finally, we study recordings of macaque monkey neurons using a probabilistic model based on recurrent neural networks. VSMC reaches the same accuracy as IWAE, but does so with less computation.

Related Work Much effort has been dedicated to learning good proposals for SMC [Cornebise, 2009]. Guarniero et al. [2017] adapt proposals through iterative refinement. Naesseth et al. [2015] uses a Monte Carlo approximation to the (locally) optimal proposal [Doucet and Johansen, 2009]. Gu et al. [2015] learn proposals by minimizing the Kullback-Leibler (KL) from the posterior to proposal using SMC samples; this strategy can suffer from high variance when the initial SMC proposal is poor. Paige and Wood [2016] learn proposals by forward simulating and inverting the model. In contrast to all these methods, VSMC optimizes the proposal directly with respect to KL divergence from the SMC sampling process to the posterior.

VSMC uses auxillary variables in a posterior approximation. This relates to work in VI, such as Hamiltonian VI [Salimans et al., 2015], variational Gaussian processes [Tran et al., 2016], hierarchical variational models [Ranganath et al., 2016b], and deep auxiliary variational auto-encoders [Maaløe et al., 2016]. Another approach uses a sequence of invertible functions to transform a simple variational approximation to a complex one [Rezende and Mohamed, 2015, Dinh et al., 2014]. All of these rich approximations can be embedded inside VSMC to build more flexible proposals.

Archer et al. [2015], Johnson et al. [2016] develop variational inference for state space models with conjugate dynamics, while Krishnan et al. [2017] develop variational approximations for models with nonlinear dynamics and additive Gaussian noise. In contrast, VSMC is agnostic to the distributional choices in the dynamics and noise.

Importance weighted auto-encoders [Burda et al., 2016] obtain the same lower bound as variational importance sampling (VIS), a special case of VSMC. However, VIS provides a new interpretation that enables a more accurate variational approximation; this relates to another interpretation of IWAE by Cremer et al. [2017], Bachman and Precup [2015]. Variational particle approximations [Saeedi et al., 2014] also provide variational approximation that improve with the number of particles, but they are restricted to discrete latent variables.

Finally, the log-marginal likelihood lower bound (6) was developed concurrently and independently by Maddison et al. [2017] and Le et al. [2017]. The difference with our work lies in how we derive the bound and the implications we explore. Maddison et al. [2017], Le et al. [2017] derive the bound using Jensen's inequality on the SMC expected log-marginal likelihood estimate, focusing on approximate marginal likelihood estimation of model parameters. Rather, we derive (6) as a tractable lower bound to the exact evidence lower bound (ELBO) for the new variational family VSMC. In addition to a lower bound on the log-marginal likelihood, this view provides a new variational approximation to the posterior.

2 Background

We begin by introducing the foundation for variational sequential Monte Carlo (VSMC). Let $p(x_{1:t}, y_{1:t})$ be a sequence of probabilistic models for latent (unobserved) $x_{1:t}$

and data $y_{1:t}$, with $t = 1, \ldots, T$. In Bayesian inference, we are interested in computing the posterior distribution $p(x_{1:T} \mid y_{1:T})$. Two concrete examples, both from the time-series literature, are hidden Markov models and state space models [Cappé et al., 2005]. In both cases, the joint density factorizes as

$$p(x_{1:T}, y_{1:T}) = f(x_1) \prod_{t=2}^{T} f(x_t \mid x_{t-1}) \prod_{t=1}^{T} g(y_t \mid x_t),$$

where f is the prior on x, and g is the observation (data) distribution. For most models computing the posterior $p(x_{1:T} \mid y_{1:T})$ is computationally intractable, and we need approximations such as VI and SMC. Here we construct posterior approximations that combine these two ideas.

In the following sections, we review variational inference and sequential Monte Carlo, develop a variational approximation based on the samples generated by SMC, and develop a tractable objective to improve the quality of the SMC variational approximation. For concreteness, we focus on the state space model above. But we emphasize that VSMC applies to any sequence of probabilistic models, just like standard SMC [Del Moral et al., 2006, Doucet and Johansen, 2009, Naesseth et al., 2014].

Variational Inference In variational inference we postulate an approximating family of distributions with variational parameters λ, $q(x_{1:T}; \lambda)$. Then we minimize a divergence, often the KL divergence, between the approximating family and the posterior so that $q(x_{1:T}; \lambda) \approx p(x_{1:T} \mid y_{1:T})$. This minimization is equivalent to maximizing the ELBO [Jordan et al., 1999],

$$\mathcal{L}(\lambda) = \mathbb{E}_{q(x_{1:T};\lambda)} \left[\log p(x_{1:T}, y_{1:T}) - \log q(x_{1:T}; \lambda) \right]. \tag{1}$$

VI turns posterior inference into an optimization problem.

Sequential Monte Carlo SMC is a sampling method designed to approximate a sequence of distributions, $p(x_{1:t} \mid y_{1:t})$ for $t = 1 \ldots T$ with special emphasis on the posterior $p(x_{1:T} \mid y_{1:T})$. For a thorough introduction to SMC see Doucet and Johansen [2009], Doucet et al. [2001], Schön et al. [2015].

To approximate $p(x_{1:t} \mid y_{1:t})$ SMC uses weighted samples,

$$p(x_{1:t} \mid y_{1:t}) \approx \widehat{p}(x_{1:t} \mid y_{1:t}) \triangleq \sum_{i=1}^{N} \frac{w_t^i}{\sum_\ell w_t^\ell} \delta_{x_{1:t}^i}, \tag{2}$$

where δ_X is the Dirac measure at X.

We construct the weighted set of particles sequentially for $t = 1, \ldots, T$. At time $t = 1$ we use standard importance sampling $x_1^i \sim r(x_1)$. For $t > 1$, we start each step by *resampling* auxiliary *ancestor variables* $a_{t-1}^i \in \{1, \ldots, N\}$ with probability proportional to the importance weights w_{t-1}^j; next we propose new values, append them to the end of the trajectory, and reweight as follows:

$$\begin{aligned}
&\textit{resample} && a_{t-1}^i \sim \text{Categorical}(w_{t-1}^j / \textstyle\sum_\ell w_{t-1}^\ell) \\
&\textit{propose} && x_t^i \sim r(x_t \mid x_{t-1}^{a_{t-1}^i}), \\
&\textit{append} && x_{1:t}^i = (x_{1:t-1}^{a_{t-1}^i}, x_t^i), \\
&\textit{reweight} && w_t^i = f(x_t^i \mid x_{t-1}^{a_{t-1}^i})\, g(y_t \mid x_t^i) / r(x_t^i \mid x_{t-1}^{a_{t-1}^i}).
\end{aligned}$$

We refer to the final particles (samples) $x_{1:T}^i$ as *trajectories*. Panels (a) and (b) of Figure 1 show sets of weighted trajectories. The size of the dots represents the weights w_t^i and the arrows represent the ancestors a_{t-1}^i. Importance sampling omits the resampling step, so each ancestor is given by the corresponding particle for the preceding time step.

The trajectories $x_{1:T}^i$ and weights w_T^i define the SMC approximation to the posterior. Critically, as we increase the number of particles, the posterior approximation becomes arbitrarily accurate. SMC also yields an unbiased estimate of the marginal likelihood,

$$\widehat{p}(y_{1:T}) = \prod_{t=1}^{T} \frac{1}{N} \sum_{i=1}^{N} w_t^i. \tag{3}$$

This estimate will play an important role in the VSMC objective.

The proposal distribution $r(x_t \mid x_{t-1})$ is the key design choice. A common choice is the model prior f—it is known as the bootstrap particle filter (BPF) [Gordon et al., 1993]. However, proposing from the prior often leads to a poor approximation for a small number of particles, especially if x_t is high-dimensional. Variational SMC addresses this shortcoming; it learns parameterized proposal distributions for efficient inference.

3 Variational Sequential Monte Carlo

We develop VSMC, a new class of variational approximations based on SMC. We first define how to sample from the VSMC family and then derive its distribution. Though generating samples is straightforward, the density is intractable. To this end, we derive a tractable objective, a new lower bound to the ELBO, that is amenable to stochastic optimization. Then, we present an algorithm to fit the variational parameters. Finally, we explore how to learn model parameters using variational expectation-maximization.

To sample from the VSMC family, we run SMC (with the proposals parameterized by variational parameters λ) and then sample once from the empirical approximation of the posterior (2). Because the proposals $r(x_t \mid x_{t-1}; \lambda)$ depend on λ, so does the SMC empirical approximation. Algorithm 1 summarizes the generative process for the VSMC family.

Algorithm 1 Variational Sequential Monte Carlo

Require: Targets $p(x_{1:t}, y_{1:t})$, proposals $r(x_t \mid x_{t-1}; \lambda)$, and number of particles N.

1: **for** $i = 1 \dots N$ **do**
2: Simulate x_1^i from $r(x_1; \lambda)$
3: Set $w_1^i = f(x_1^i)\, g(y_1 \mid x_1^i) / r(x_1^i; \lambda)$
4: **end for**
5: **for** $t = 2 \dots T$ **do**
6: **for** $i = 1 \dots N$ **do**
7: Simulate a_{t-1}^i with $\Pr(a_{t-1}^i = j) = \frac{w_{t-1}^j}{\sum_\ell w_{t-1}^\ell}$
8: Simulate x_t^i from $r(x_t \mid x_{t-1}^{a_{t-1}^i}; \lambda)$
9: Set $x_{1:t}^i = (x_{1:t-1}^{a_{t-1}^i}, x_t^i)$
10: Set $w_t^i = f(x_t^i \mid x_{t-1}^{a_{t-1}^i})\, g(y_t \mid x_t^i) / r(x_t^i \mid x_{t-1}^{a_{t-1}^i}; \lambda)$
11: **end for**
12: **end for**
13: Simulate b_T with $\Pr(b_T = j) = w_T^j / \sum_\ell w_T^\ell$
14: **return** $x_{1:T} \triangleq x_{1:T}^{b_T}$

The variational distribution $q(x_{1:T}; \lambda)$ marginalizes out all the variables produced in the sampling process, save for the output sample $x_{1:T}$. This marginal comes from the joint distribution of all variables generated by VSMC,

$$\widetilde{\phi}(x_{1:T}^{1:N}, a_{1:T-1}^{1:N}, b_T; \lambda) = \underbrace{\left[\prod_{i=1}^N r(x_1^i; \lambda)\right]}_{\text{step 2}} \cdot \prod_{t=2}^T \prod_{i=1}^N \underbrace{\left[\frac{w_{t-1}^{a_{t-1}^i}}{\sum_\ell w_{t-1}^\ell}\right]}_{\text{step 7}} \underbrace{r(x_t^i \mid x_{t-1}^{a_{t-1}^i}; \lambda)}_{\text{step 8}} \underbrace{\left[\frac{w_T^{b_T}}{\sum_\ell w_T^\ell}\right]}_{\text{step 13}}. \quad (4)$$

(We have annotated this equation with the steps from the algorithm.) In this joint, the final output sample is defined by extracting the b_T-th trajectory $x_{1:T} = x_{1:T}^{b_T}$. Note that the data $y_{1:T}$ enter via the weights and (optionally) the proposal distribution. This joint density is easy to calculate, but for variational inference we need the marginal distribution of $x_{1:T}$. We derive this next.

Let $b_t \triangleq a_t^{b_{t+1}}$ for $t \leq T-1$ denote the ancestors for the trajectory $x_{1:T}$ returned by Algorithm 1. Furthermore, let $\neg b_{1:T}$ be all particle indices *not* equal to $(b_1, \dots, b_T)$, i.e. exactly all the particles that were not returned by Algorithm 1. Then the marginal distribution of $x_{1:T} = x_{1:T}^{b_{1:T}} = (x_1^{b_1}, x_2^{b_2}, \dots, x_T^{b_T})$ is given by the following proposition.

Proposition 1. *The* VSMC *approximation on* $x_{1:T}$ *is*

$$q(x_{1:T} \mid y_{1:T}; \lambda) = p(x_{1:T}, y_{1:T})\, \mathbb{E}_{\widetilde{\phi}\left(x_{1:T}^{\neg b_{1:T}}, a_{1:T-1}^{\neg b_{1:T-1}}; \lambda\right)} \left[\widehat{p}(y_{1:T})^{-1}\right]. \quad (5)$$

Proof. See the supplementary material A.1. □

This has an intuitive form: the density of the variational posterior is equal to the exact joint times the expected inverse of the normalization constant (c.f. (3)). While we can estimate this expectation with Monte Carlo, it yields a biased estimate of $\log q(x_{1:T} \mid y_{1:T}; \lambda)$ and the ELBO (1).

The surrogate ELBO. To derive a tractable objective, we develop a lower bound to the ELBO that is also amenable to stochastic optimization. It is

$$\widetilde{\mathcal{L}}(\lambda) \triangleq \sum_{t=1}^T \mathbb{E}_{\widetilde{\phi}(x_{1:t}^{1:N}, a_{1:t-1}^{1:N}; \lambda)} \left[\log\left(\frac{1}{N}\sum_{i=1}^N w_t^i\right)\right] = \mathbb{E}\left[\log \widehat{p}(y_{1:T})\right] \quad (6)$$

We call $\widetilde{\mathcal{L}}(\lambda)$ the *surrogate* ELBO. It is a lower bound to the true ELBO for VSMC or, equivalently, an upper bound on the KL divergence. The following theorem formalizes this fact:

Theorem 1 (Surrogate ELBO)**.** *The surrogate* ELBO (6)*, is a lower bound to the* ELBO (1) *when* q *is defined by* (5)*, i.e.*

$$\log p(y_{1:T}) \geq \mathcal{L}(\lambda) \geq \widetilde{\mathcal{L}}(\lambda).$$

Proof. See the supplementary material A.2. □

The surrogate ELBO is the expected SMC log-marginal likelihood estimate. We can estimate it unbiasedly as a byproduct of sampling from the VSMC variational approximation (Algorithm 1). We run the algorithm and use the estimate to perform stochastic optimization of the surrogate ELBO.

Stochastic Optimization. While the expectations in the surrogate ELBO are still not available in closed form, we can estimate it and its gradients with Monte Carlo. This admits a stochastic optimization algorithm for finding the optimal variational parameters of the VSMC family.

We assume the proposals $r(x_t \mid x_{t-1}; \lambda)$ are reparameterizable, i.e., we can simulate from r by setting $x_t = h(x_{t-1}, \varepsilon_t; \lambda)$, $\varepsilon_t \sim s(\varepsilon_t)$ for some distribution s not a function of λ. With this assumption, rewrite the gradient of (6) by using the reparameterization trick [Kingma and Welling, 2014, Rezende et al., 2014],

$$\nabla \widetilde{\mathcal{L}}(\lambda) = g_{\text{rep}} + g_{\text{score}} \quad (7)$$
$$g_{\text{rep}} = \mathbb{E}\left[\nabla \log \widehat{p}(y_{1:T})\right],$$
$$g_{\text{score}} = \mathbb{E}\left[\log \widehat{p}(y_{1:T}) \nabla \log \widetilde{\phi}(a_{1:T-1}^{1:N} \mid \varepsilon_{1:T}^{1:N}; \lambda)\right].$$

This expansion follows from the product rule, just as in the generalized reparameterizations of Ruiz et al. [2016] and Naesseth et al. [2017]. Note that all x_t^i, implicit in the weights w_t^i and $\widehat{p}(y_{1:T})$ are now replaced with their reparameterizations $h(\cdot; \lambda)$. The ancestor variables are discrete and cannot be reparameterized—this can lead to high variance in the score function term, g_{score} from (7).

In Section 5, we empirically assess the impact of ignoring g_{score} for optimization. We empirically study optimizing with and without the score function term for a small state space model where standard variance reduction techniques, explained below, are sufficient. We lower the variance using Rao-Blackwellization [Robert and Casella, 2004, Ranganath et al., 2014], noting that the ancestor variables a_{t-1} have no effect on weights prior to time t,

$$g_{\text{score}} = \sum_{t=2}^{T} \mathbb{E}\left[\log \frac{\widehat{p}(y_{1:T})}{\widehat{p}(y_{1:t-1})} \left(\sum_{i=1}^{N} \nabla \log \frac{w_{t-1}^{a_{t-1}^i}}{\sum_\ell w_{t-1}^\ell}\right)\right]. \quad (8)$$

Furthermore, we use the score function $\nabla \log \widetilde{\phi}(a_{1:T-1}^{1:N} \,|\, \varepsilon_{1:T}^{1:N}\,; \lambda)$ with an estimate of the future log average weights as a control variate [Ranganath et al., 2014].

We found that ignoring the score function term g_{score} (8) from the ancestor variables, leads to faster convergence and very little difference in final ELBO. This corresponds to approximating the gradient of $\widetilde{\mathcal{L}}$ by

$$\nabla \widetilde{\mathcal{L}}(\lambda) \approx \mathbb{E}\left[\nabla \log \widehat{p}(y_{1:T})\right] = g_{\text{rep}}. \quad (9)$$

This is the gradient we propose to use for optimizing the variational parameters of VSMC. See the supplementary material A.3 for more details, where we also provide a general score function-like estimator and the control variates.

Algorithm. We now describe the full algorithm to optimize the VSMC variational approximation. We form stochastic gradients $\widehat{\nabla}\widetilde{\mathcal{L}}(\lambda)$ by estimating (9) using a single sample from $s(\cdot)\widetilde{\phi}(\cdot \,|\, \cdot\,; \lambda)$. The sample is obtained as a byproduct of sampling VSMC (Algorithm 1). We use the step-size sequence Adam [Kingma and Ba, 2015] or ρ^n proposed by Kucukelbir et al. [2017],

$$\rho^n = \eta \cdot n^{-1/2+\delta} \cdot \left(1 + \sqrt{s^n}\right)^{-1},$$
$$s^n = t\left(\widehat{\nabla}\widetilde{\mathcal{L}}(\lambda^n)\right)^2 + (1-t)s^{n-1}, \quad (10)$$

where n is the iteration number. We set $\delta = 10^{-16}$ and $t = 0.1$, and we try different values for η. Algorithm 2 summarizes this optimization algorithm.[1]

Variational Expectation Maximization. Suppose the target distribution of interest $p(x_{1:T} \,|\, y_{1:T}\,; \theta)$ has a set of unknown parameters θ. We can fit the parameters using variational expectation-maximization (VEM) [Beal and Ghahramani, 2003]. The surrogate ELBO is updated accordingly

$$\log p(y_{1:T}\,; \theta) \geq \tilde{\mathcal{L}}(\lambda, \theta) \quad (11)$$

[1]Reference implementation using Adam is available at github.com/blei-lab/variational-smc.

Algorithm 2 Stochastic Optimization for VSMC

Require: Data $y_{1:T}$, model $p(x_{1:T}, y_{1:T})$, proposals $r(x_t \,|\, x_{t-1}\,; \lambda)$, number of particles N
Ensure: Variational parameters $\lambda^\star$
1: **repeat**
2: Estimate the gradient $\widehat{\nabla}\widetilde{\mathcal{L}}(\lambda^n)$ given by (9)
3: Compute stepsize ρ^n with (10)
4: Update $\lambda^{n+1} = \lambda^n + \rho^n \widehat{\nabla}\widetilde{\mathcal{L}}(\lambda^n)$
5: **until convergence**

where the normalization constant $p(y_{1:T}\,; \theta)$ is now a function of the parameters θ. Note that the expression for $\tilde{\mathcal{L}}(\lambda, \theta)$ is exactly the same as (6), but where the weights (and potentially proposals) now include a dependence on the model parameters θ. Analogously, the reparameterization gradients have the same form as (9). We can maximize (11), with respect to both θ and λ, using stochastic optimization. With data subsampling, VSMC extends to large-scale datasets of conditionally independent sequences [Hoffman et al., 2013, Titsias and Lázaro-Gredilla, 2014].

4 Perspectives on Variational SMC

We give some perspectives on VSMC. First, we consider the VSMC special cases of $N = 1$ and $T = 1$. For $N = 1$, VSMC reduces to a structured variational approximation: there is no resampling and the variational distribution is exactly the proposal. For $T = 1$, VSMC leads to a special case we call variational importance sampling, and a reinterpretation of the IWAE [Burda et al., 2016], which we explore further in the first half of this section.

Then, we think of sampling from VSMC as sampling a highly optimized SMC approximation. This means many of the theoretical SMC results developed over the past 25 years can be adapted for VSMC. We explore some examples in the second half of this section.

Variational Importance Sampling (VIS). The case where $T = 1$ is SMC without any resampling, i.e., importance sampling. The corresponding special case of VSMC is VIS. The surrogate ELBO for VIS is exactly equal to the IWAE lower bound [Burda et al., 2016].

This equivalence provides new intuition behind the IWAE's variational approximation on the latent variables. If we want to make use of the approximation $q(x_{1:T}\,; \lambda^\star)$ learned with the IWAE lower bound, samples from the latent variables should be generated with Algorithm 1, i.e. VIS. For VIS it is possible to show that the surrogate ELBO is always tighter than the one obtained by standard VB (equivalent to VIS with $N = 1$) [Burda et al., 2016]. This result does not carry over to VSMC, i.e. we can find cases when the resampling creates a looser bound compared to standard VB or VIS. However, in practice the VSMC lower bound outperforms

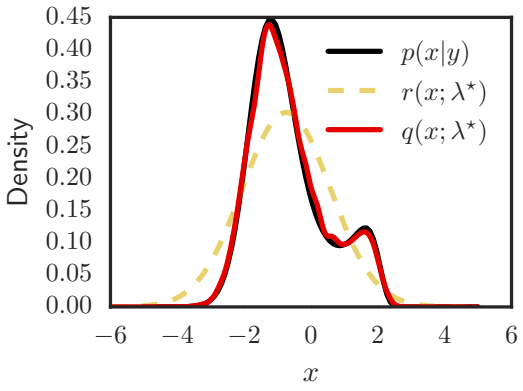


Figure 2: Example of VIS $q(x\,;\lambda)$ approximating a multimodal $p(x \mid y)$ with a Gaussian proposal $r(x\,;\lambda)$.

the VIS lower bound.

Figure 2 provides a simple example of VIS applied to a multimodal $p(x \mid y) \propto \mathcal{N}(x\,;0,1)\,\mathcal{N}(y\,;x^2/2, e^{x/2})$ with a normal proposal $r(x\,;\lambda) = \mathcal{N}(x\,;\mu,\sigma^2)$ and a kernel density estimate of the corresponding variational approximation $q(x\,;\lambda)$. The number of particles is $N = 10$. Standard VB with a Gaussian approximation only captures one of the two modes; which one depends on the initialization. We see that even a simple proposal can lead to a very flexible posterior approximation. This property is also inherited by the more general $T > 1$ case, VSMC.

Theoretical Properties. The normalization constant estimate of the SMC sampler, $\widehat{p}(y_{1:T})$, is unbiased [Del Moral, 2004, Pitt et al., 2012, Naesseth et al., 2014]. This, together with Jensen's inequality, implies that the surrogate ELBO $\mathbb{E}[\log \widehat{p}(y_{1:T})]$ is a lower bound to $\log p(y_{1:T})$. If $\log \widehat{p}(y_{1:T})$ is uniformly integrable it follows [Del Moral, 2004], as $N \to \infty$, that

$$\widetilde{\mathcal{L}}(\lambda) = \mathcal{L}(\lambda) = \log p(y_{1:T}).$$

This fact means that the gap in Theorem 1 disappears and the distribution of the trajectory returned by VSMC will tend to the true target distribution $p(x_{1:T} \mid y_{1:T})$. A bound on the KL divergence gives us the rate

$$\mathrm{KL}\left(q(x_{1:T}\,;\lambda) \,\middle\|\, p(x_{1:T} \mid y_{1:T})\right) \leq \frac{c(\lambda)}{N},$$

for some constant $c(\lambda) < \infty$. This is a special case of a "propagation of chaos" result from Del Moral [2004, Theorem 8.3.2].

We can arrive at this result informally by studying (5): as the number of particles increases, the marginal likelihood estimate will converge to the true marginal likelihood and the variational posterior will converge to the true posterior. Huggins and Roy [2017] provide further bounds on various divergences and metrics between SMC and the target distribution.

VSMC and T. Like SMC, variational sequential Monte Carlo scales well with T. Bérard et al. [2014] show a central limit theorem for the SMC approximation $\log \widehat{p}(y_{1:T}) - \log p(y_{1:T})$ with $N = bT$, where $b > 0$, as $T \to \infty$. Under the same conditions as in that work, and assuming that $\log \widehat{p}(y_{1:T})$ is uniformly integrable, we can show that

$$\mathrm{KL}\left(q(x_{1:T}\,;\lambda) \,\middle\|\, p(x_{1:T} \mid y_{1:T})\right) \leq -\mathbb{E}\left[\log \frac{\widehat{p}(y_{1:T})}{p(y_{1:T})}\right] \xrightarrow[T\to\infty]{} \frac{\sigma^2(\lambda)}{2b}, \quad 0 < \sigma^2(\lambda) < \infty.$$

The implication for VSMC is significant. We can make the variational approximation *arbitrarily accurate* by setting $N \propto T$, even as T goes to infinity. The supplement shows that this holds in practice; see A.4 for the toy example from Figure 1. We emphasize that neither standard VB nor IWAE (VIS) have this property.

5 Empirical Study

Linear Gaussian State Space Model The linear Gaussian SSM is a ubiquitous model of time series data that enjoys efficient algorithms for computing the exact posterior. We use this model to study the convergence properties and impact of biased gradients for VSMC. We further use it to confirm that we learn good proposals. We compare to the bootstrap particle filter (BPF), which uses the prior as a proposal, and the (locally) optimal proposal that tilts the prior with the likelihood.

The model is

$$x_t = Ax_{t-1} + v_t,$$
$$y_t = Cx_t + e_t,$$

where $v_t \sim \mathcal{N}(0, Q)$, $e_t \sim \mathcal{N}(0, R)$, and $x_1 \sim \mathcal{N}(0, I)$. The log-marginal likelihood $\log p(y_{1:T})$ can be computed using the Kalman filter.

We study the impact of the biased gradient (9) for optimizing the surrogate ELBO (6). First, consider a simple scalar model with $A = 0.5$, $Q = 1$, $C = 1$, $R = 1$, and $T = 2$. For the proposal we use $r(x_t \mid x_{t-1}\,;\lambda) = \mathcal{N}(x_t\,;\lambda + 0.5x_{t-1}, 1)$, with $x_0 \equiv 0$. Figure 3 (left) shows the mean and spread of estimates of g_{score} (8), with control variates, and g_{rep} (9), as a function of λ for four randomly generated datasets. The optimal setting of λ is where the sum of the means is equal to zero. Ignoring the score function term g_{score} (8) will lead to a perturbation of the optimal λ. However, even for this simple model, the variance of the score function term (red) is several orders of magnitude higher than that of the reparameterization term (blue), despite the variance reduction techniques of Section 3. This variance has a significant impact on the convergence speed of the stochastic optimization.

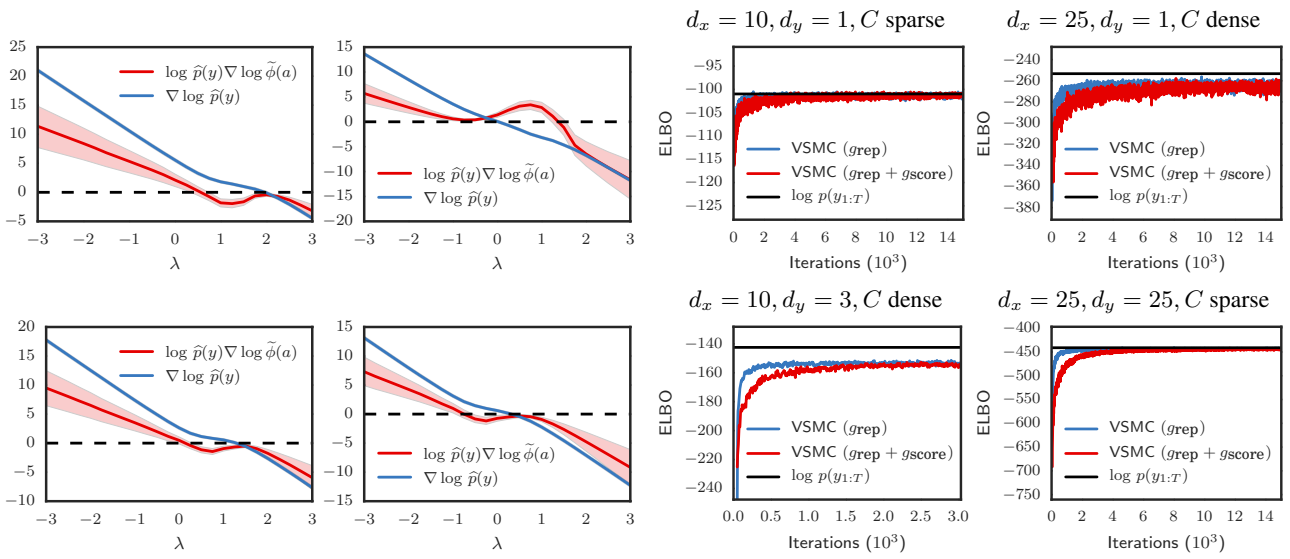


Figure 3: *(Left)* Mean and spread of the stochastic gradient components g_{score} (8) and g_{rep} (9), for the scalar linear Gaussian model on four randomly generated datasets, where the number of particles is $N = 2$. *(Right)* Log-marginal likelihood ($\log p(y_{1:T})$) and ELBO as a function of iterations for VSMC with biased gradients (blue) or unbiased gradients (red). Results for four different linear Gaussian models.

Next, we study the magnitude of the perturbation, and its effect on the surrogate ELBO. We generate data with $T = 10$, $(A)_{ij} = \alpha^{|i-j|+1}$ for $\alpha = 0.42$, $Q = I$, and $R = I$. We explored several settings of $d_x = \dim(x_t)$, $d_y = \dim(y_t)$, and C. Sparse C measures the first d_y components of x_t, and dense C has randomly generated elements $C_{ij} \sim \mathcal{N}(0, 1)$. Figure 3 (right) shows the true log-marginal likelihood and ELBO as a function of iteration. It shows VSMC with biased gradients (blue) and unbiased gradients (red). We choose the proposal

$$r(x_t \mid x_{t-1}; \lambda) = \mathcal{N}\left(x_t \mid \mu_t + \operatorname{diag}(\beta_t) A x_{t-1}, \operatorname{diag}(\sigma_t^2)\right).$$

with $\lambda = \{\mu_t, \beta_t, \sigma_t^2\}_{t=1}^T$, and set the number of particles to $N = 4$. Note that while the gradients are biased, the resulting ELBO is not. We can see that the final VSMC ELBO values are very similar, regardless of whether we train with biased or unbiased gradients. However, biased gradients converge faster. Thus, we use biased gradients in the remainder of our experiments.

Next, we study the effect of learning the proposal using VSMC compared with standard proposals in the SMC literature. The most commonly used is the BPF, sampling from the prior f. We also consider the so-called optimal proposal, $r \propto f \cdot g$, which minimizes the variance of the incremental importance weights [Doucet and Johansen, 2009]. Table 1 shows results for a linear Gaussian SSM when $T = 25$, $Q = 0.1^2 I$, $R = 1$, $d_x = 10$, and $d_y = 1$. Because of the relatively high-dimensional state, BPF exhibits significant bias whereas the optimal proposal SMC performs much better. VSMC outperforms them both, learning an accurate proposal that results in an ELBO only 0.9 nats lower than the true log-marginal likelihood. We further emphasize that the optimal proposal is unavailable for most models.

Table 1: ELBO for BPF, SMC with (locally) optimal proposal, and VSMC. The true log-marginal likelihood is given by $\log p(y_{1:T}) = -236.9$.

	BPF	Optimal SMC	VSMC
ELBO	-6701.4	-253.4	$\mathbf{-237.8}$

Stochastic Volatility A common model in financial econometrics is the (multivariate) stochastic volatility model [Chib et al., 2009]. The model is

$$x_t = \mu + \phi(x_{t-1} - \mu) + v_t,$$
$$y_t = \beta \exp\left(\frac{x_t}{2}\right) e_t,$$

where $v_t \sim \mathcal{N}(0, Q)$, $e_t \sim \mathcal{N}(0, I)$, $x_1 \sim \mathcal{N}(\mu, Q)$, and $\theta = (\mu, \phi, Q, \beta)$. (In the multivariate case, multiplication is element-wise.) Computing $\log p(y_{1:T}; \theta)$ and its gradients for this model is intractable, we study the VEM approximation to find the unknown parameters θ. We compare VSMC with IWAE and structured VI. For the proposal in VSMC and IWAE we choose

$$r(x_t \mid x_{t-1}; \lambda, \theta) \propto f(x_t \mid x_{t-1}; \theta)\, \mathcal{N}(x_t; \mu_t, \Sigma_t),$$

with variational parameters $\lambda = (\mu_1, \ldots, \mu_T, \Sigma_1, \ldots, \Sigma_T)$. We define the variational approximation for structured VI to be $q(x_{1:T}; \lambda, \theta) = \prod_{t=1}^T r(x_t \mid x_{t-1}; \lambda, \theta)$.

We study 10 years of monthly returns (9/2007 to 8/2017) for the exchange rate of 22 international currencies with

Table 2: ELBO for the stochastic volatility model with $T = 119$ on exchange rate data. We compare VSMC (this paper) with IWAE and structured VI.

	Method	ELBO
	Structured VI	6905.1
$N = 4$	IWAE	6911.2
	VSMC	**6921.6**
$N = 8$	IWAE	6912.4
	VSMC	**6935.8**
$N = 16$	IWAE	6913.3
	VSMC	**6936.6**

respect to US dollars. The data is from the Federal Reserve System. Table 2 reports the optimized ELBO (higher is better) for different settings of the number of particles/samples $N = \{4, 8, 16\}$. VSMC outperforms the competing methods with almost 0.2 nats per time-step.

In theory we can improve the bound of both IWAE and VSMC by increasing the number of samples N. This means we can first learn proposals using only a few particles N, for computational efficiency. Then, at test time, we can increase N as needed for improved accuracy. We study the impact of increasing the number of samples for VSMC and IWAE using fix $\theta^\star$ and $\lambda^\star$ optimized with $N = 16$. Figure 4 shows that the gain for IWAE is limited, whereas for VSMC it can be significant.

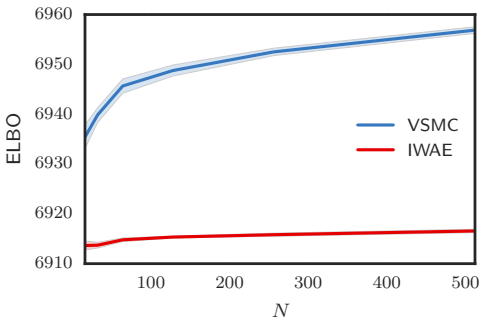


Figure 4: The estimated ELBO for VSMC (this paper) and IWAE , with confidence bands, as a function of the number of particles N for fix $\theta^\star$, $\lambda^\star$.

Deep Markov Model An important problem in neuroscience is understanding dynamics of neural circuits. We study a population of 105 motor cortex neurons simultaneously recorded in a macaque monkey as it performed reaching movements [c.f. Gao et al., 2016]. In each trial, the monkey reached toward one of fourteen targets; each trial is $T = 21$ time steps long. We train on 700 trials and test on 84.

We use recurrent neural networks to model both the dynamics and observations. The model is

$$x_t = \mu_\theta(x_{t-1}) + \exp\left(\sigma_\theta(x_{t-1})/2\right) v_t,$$
$$y_t \sim \text{Poisson}\left(\exp\left(\eta_\theta(x_t)\right)\right),$$

where $v_t \sim \mathcal{N}(0, I)$, $x_0 \equiv 0$, and μ, σ, η are neural networks parameterized by θ. The multiplication in the transition dynamics is element-wise. This is a deep Markov model [Krishnan et al., 2017].

For inference we use the following proposal for both VSMC and IWAE,

$$r(x_t \mid x_{t-1}, y_t; \lambda) \propto \mathcal{N}\left(x_t; \mu^x_\lambda(x_{t-1}), \exp\left(\sigma^x_\lambda(x_{t-1})\right)\right) \times \mathcal{N}\left(x_t; \mu^y_\lambda(y_t), \exp\left(\sigma^y_\lambda(y_t)\right)\right),$$

where $\mu^x, \sigma^x, \mu^y, \sigma^y$ are neural networks parameterized by λ, and the proposal factorizes over the components of x_t. Figure 5 illustrates the result for $d_x = \{3, 5, 10\}$ with $N = 8$. VSMC gets to the same ELBO faster.

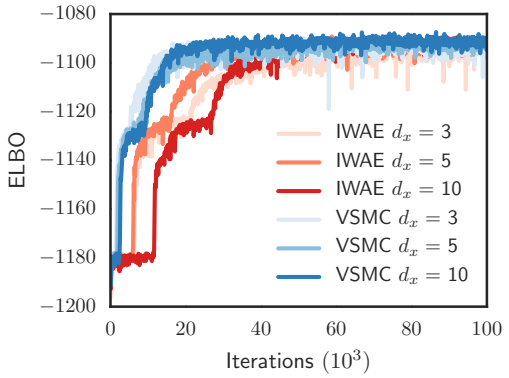


Figure 5: The estimated ELBO of the neural population test data as a function of iterations for VSMC (this paper) and IWAE, for $d_x = \{3, 5, 10\}$ and $T = 21$.

6 Conclusions

We introduced the variational sequential Monte Carlo (VSMC) family, a new variational approximating family that provides practitioners with a flexible, accurate, and powerful approximate Bayesian inference algorithm. VSMC melds variational inference (VI) and sequential Monte Carlo (SMC). This results in a variational approximation that lets us trade-off fidelity to the posterior with computational complexity.

Acknowledgements

Christian A. Naesseth is supported by CADICS, a Linnaeus Center, funded by the Swedish Research Council (VR). Scott W. Linderman is supported by the Simons Foundation SCGB-418011. This work is supported by ONR N00014-11-1-0651, DARPA PPAML FA8750-14-2-0009, the Alfred P. Sloan Foundation, and the John Simon Guggenheim Foundation.

References

E. Archer, I. Memming Park, L. Buesing, J. Cunningham, and L. Paninski. Black box variational inference for state space models. *arXiv:1511.07367*, Nov. 2015.

P. Bachman and D. Precup. Training deep generative models: Variations on a theme. In *NIPS Approximate Inference Workshop*, 2015.

M. J. Beal and Z. Ghahramani. The variational Bayesian EM algorithm for incomplete data: with application to scoring graphical model structures. *Bayesian statistics*, 2003.

Y. Burda, R. Grosse, and R. Salakhutdinov. Importance weighted autoencoders. In *International Conference on Learning Representations*, 2016.

J. Bérard, P. Del Moral, and A. Doucet. A lognormal central limit theorem for particle approximations of normalizing constants. *Electronic Journal of Probability*, 2014.

O. Cappé, E. Moulines, and T. Rydén. *Inference in Hidden Markov Models*. Springer-Verlag New York, 2005.

S. Chib, Y. Omori, and M. Asai. *Multivariate Stochastic Volatility*, pages 365–400. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

J. Cornebise. *Adaptive Sequential Monte Carlo Methods*. PhD thesis, University Pierre and Marie Curie–Paris 6, 2009.

C. Cremer, Q. Morris, and D. Duvenaud. Reinterpreting importance-weighted autoencoders. *arXiv:1704.02916*, Apr. 2017.

P. Del Moral. *Feynman-Kac Formulae: Genealogical and interacting particle systems with applications*. Springer-Verlag New York, 2004.

P. Del Moral, A. Doucet, and A. Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2006.

L. Dinh, D. Krueger, and Y. Bengio. Nice: Non-linear independent components estimation. *arXiv:1410.8516*, 2014.

A. Doucet and A. M. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of nonlinear filtering*, 12(656-704):3, 2009.

A. Doucet, N. De Freitas, and N. Gordon. An introduction to sequential Monte Carlo methods. In *Sequential Monte Carlo methods in practice*. Springer-Verlag New York, 2001.

Y. Gao, E. W. Archer, L. Paninski, and J. P. Cunningham. Linear dynamical neural population models through nonlinear embeddings. In *Advances in Neural Information Processing Systems*, 2016.

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

S. Gu, Z. Ghahramani, and R. E. Turner. Neural adaptive sequential Monte Carlo. In *Advances in Neural Information Processing Systems*, 2015.

P. Guarniero, A. M. Johansen, and A. Lee. The iterated auxiliary particle filter. *Journal of the American Statistical Association*, 112(520):1636–1647, 2017.

M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14:1303–1347, May 2013.

J. H. Huggins and D. M. Roy. Sequential Monte Carlo as approximate sampling: bounds, adaptive resampling via ∞-ESS, and an application to particle Gibbs. *arXiv:1503.00966v2*, Mar. 2017.

M. Johnson, D. K. Duvenaud, A. Wiltschko, R. P. Adams, and S. R. Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in Neural Information Processing Systems*, 2016.

M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, Nov. 1999.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

D. P. Kingma and M. Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.

G. Kitagawa. Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of computational and graphical statistics*, 5(1):1–25, 1996.

R. Krishnan, U. Shalit, and D. Sontag. Structured inference networks for nonlinear state space models. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

A. Kucukelbir, D. Tran, R. Ranganath, A. Gelman, and D. M. Blei. Automatic differentiation variational inference. *Journal of Machine Learning Research*, 18(1): 430–474, Jan. 2017. ISSN 1532-4435.

T. A. Le, M. Igl, T. Jin, T. Rainforth, and F. Wood. Auto-Encoding Sequential Monte Carlo. *arXiv:1705.10306*, May 2017.

L. Maaløe, C. K. Sønderby, S. K. Sønderby, and O. Winther. Auxiliary deep generative models. In *International Conference on Machine Learning*, 2016.

C. J. Maddison, D. Lawson, G. Tucker, N. Heess, M. Norouzi, A. Mnih, A. Doucet, and Y. Whye Teh. Filtering variational objectives. In *Advances in Neural Information Processing Systems*, 2017.

C. A. Naesseth, F. Lindsten, and T. B. Schön. Sequential Monte Carlo for graphical models. In *Advances in Neural Information Processing Systems*, 2014.

C. A. Naesseth, F. Lindsten, and T. B. Schön. Nested sequential Monte Carlo methods. In *International Conference on Machine Learning*, 2015.

C. A. Naesseth, F. J. R. Ruiz, S. W. Linderman, and D. M. Blei. Reparameterization gradients through acceptance-rejection sampling algorithms. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, 2017.

B. Paige and F. Wood. Inference networks for sequential Monte Carlo in graphical models. In *International Conference on Machine Learning*, 2016.

M. K. Pitt, R. dos Santos Silva, P. Giordani, and R. Kohn. On some properties of Markov chain Monte Carlo simulation methods based on the particle filter. *Journal of Econometrics*, 2012.

R. Ranganath, S. Gerrish, and D. M. Blei. Black box variational inference. In *Artificial Intelligence and Statistics*, 2014.

R. Ranganath, A. Perotte, N. Elhadad, and D. Blei. Deep survival analysis. In *Proceedings of the 1st Machine Learning for Healthcare Conference*, pages 101–114, 2016a.

R. Ranganath, D. Tran, and D. M. Blei. Hierarchical variational models. In *International Conference on Machine Learning*, 2016b.

D. J. Rezende and S. Mohamed. Variational inference with normalizing flows. In *International Conference on Machine Learning*, 2015.

D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*, 2014.

C. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer, 2004.

F. J. R. Ruiz, M. K. Titsias, and D. M. Blei. The generalized reparameterization gradient. In *Advances in Neural Information Processing Systems*, 2016.

A. Saeedi, T. D. Kulkarni, V. Mansinghka, and S. Gershman. Variational particle approximations. *arXiv preprint arXiv:1402.5715*, 2014.

T. Salimans, D. P. Kingma, and M. Welling. Markov chain Monte Carlo and variational inference: Bridging the gap. In *International Conference on Machine Learning*, 2015.

T. B. Schön, F. Lindsten, J. Dahlin, J. Wågberg, C. A. Naesseth, A. Svensson, and L. Dai. Sequential Monte Carlo methods for system identification. In *Proceedings of the 17th IFAC Symposium on System Identification (SYSID)*, Oct 2015.

L. Stewart and P. McCarty, Jr. Use of Bayesian belief networks to fuse continuous and discrete information for target recognition, tracking, and situation assessment. In *Proc. SPIE*, volume 1699, pages 177–185, 1992.

M. Titsias and M. Lázaro-Gredilla. Doubly stochastic variational Bayes for non-conjugate inference. In *Proceedings of the 31st International Conference on Machine Learning*, 2014.

D. Tran, R. Ranganath, and D. M. Blei. The variational Gaussian process. In *International Conference on Learning Representations*, 2016.

A Variational Sequential Monte Carlo – Supplementary Material

A.1 Proof of Proposition 1

We start by noting that the distribution of all random variables generated by the VSMC algorithm is given by

$$\widetilde{\phi}(x_{1:T}^{1:N}, a_{1:T-1}^{1:N}, b_T\,;\lambda) = \frac{w_T^{b_T}}{\sum_\ell w_T^\ell}\prod_{i=1}^N r(x_1^i\,;\lambda)\cdot\prod_{t=2}^T\prod_{i=1}^N \frac{w_{t-1}^{a_{t-1}^i}}{\sum_\ell w_{t-1}^\ell} r(x_t^i|x_{t-1}^{a_{t-1}^i}\,;\lambda). \tag{12}$$

We are interested in the marginal distribution $q(x_{1:T}\,;\lambda) \triangleq \widetilde{\phi}(x_{1:T}\,;\lambda) = \mathbb{E}_{b_{1:T}}[\widetilde{\phi}(x_{1:T}^{b_{1:T}}, b_{1:T}\,;\lambda)]$. A key observation is that the distribution of $b_{1:T} \mid x_{1:T}$, the conditional distribution of the ancestral path of the returned particle, is uniform on $\{1,\dots,N\}^T$. Thus we get

$$q(x_{1:T}\,;\lambda) = \frac{\widetilde{\phi}(x_{1:T}^{b_{1:T}}, b_{1:T}\,;\lambda)}{\widetilde{\phi}(b_{1:T}\mid x_{1:T}\,;\lambda)} = \frac{1}{N^{-T}}\sum_{a_{1:T-1}^{\neg b_{1:T-1}}}\int \widetilde{\phi}(x_{1:T}^{b_{1:T}}, x_{1:T}^{\neg b_{1:T}}, a_{1:T-1}^{\neg b_{1:T-1}}\,;\lambda)\,\mathrm{d}x_{1:T}^{\neg b_{1:T}}, \tag{13}$$

where

$$\begin{aligned}
&\frac{1}{N^{-T}}\widetilde{\phi}(x_{1:T}^{b_{1:T}}, x_{1:T}^{\neg b_{1:T}}, a_{1:T-1}^{\neg b_{1:T-1}}\,;\lambda)\\
&= N^T\frac{w_1^{b_1}}{\sum_\ell w_1^\ell} r(x_1^{b_1}\,;\lambda)\prod_{t=2}^T \frac{w_t^{b_t}}{\sum_\ell w_t^\ell} r(x_t^{b_t}\mid x_{t-1}^{b_{t-1}}\,;\lambda)\cdot\prod_{\substack{i=1\\ i\neq b_1}}^N r(x_1^i\,;\lambda)\cdot\prod_{t=2}^T\prod_{\substack{i=1\\ i\neq b_t}}^N \frac{w_{t-1}^{a_{t-1}^i}}{\sum_\ell w_{t-1}^\ell} r(x_t^i|x_{t-1}^{a_{t-1}^i}\,;\lambda)\\
&= p(x_1^{b_1}, y_1)\prod_{t=2}^T \frac{p(x_{1:t}^{b_{1:t}}, y_{1:t})}{p(x_{1:t-1}^{b_{1:t-1}}, y_{1:t-1})}\prod_{t=1}^T \frac{1}{\frac{1}{N}\sum_\ell w_t^\ell}\cdot\prod_{\substack{i=1\\ i\neq b_1}}^N r(x_1^i\,;\lambda)\cdot\prod_{t=2}^T\prod_{\substack{i=1\\ i\neq b_t}}^N \frac{w_{t-1}^{a_{t-1}^i}}{\sum_\ell w_{t-1}^\ell} r(x_t^i|x_{t-1}^{a_{t-1}^i}\,;\lambda)\\
&= p(x_{1:T}^{b_{1:T}}, y_{1:T})\prod_{t=1}^T \frac{1}{\frac{1}{N}\sum_\ell w_t^\ell}\cdot\widetilde{\phi}(x_{1:T}^{\neg b_{1:T}}, a_{1:T-1}^{\neg b_{1:T-1}}\,;\lambda).
\end{aligned}$$

We insert the above expression in (13) and we get

$$\begin{aligned}
q(x_{1:T}\,;\lambda) &= p(x_{1:T}^{b_{1:T}}, y_{1:T})\sum_{a_{1:T-1}^{\neg b_{1:T-1}}}\int\left(\prod_{t=1}^T\frac{1}{N}\sum_{i=1}^N w_t^i\right)^{-1}\cdot\widetilde{\phi}(x_{1:T}^{\neg b_{1:T}}, a_{1:T-1}^{\neg b_{1:T-1}}\,;\lambda)\,\mathrm{d}x_{1:T}^{\neg b_{1:T}}\\
&= p(x_{1:T}^{b_{1:T}}, y_{1:T})\mathbb{E}_{\widetilde{\phi}(x_{1:T}^{\neg b_{1:T}}, a_{1:T-1}^{\neg b_{1:T-1}}\,;\lambda)}\left[\left(\prod_{t=1}^T\frac{1}{N}\sum_{i=1}^N w_t^i\right)^{-1}\right].
\end{aligned} \tag{14}$$

□

A.2 Proof of Theorem 1

The evidence lower bound (ELBO), using the above result about the distribution of $q(x_{1:T}\,;\lambda)$, is given by

$$\begin{aligned}
\mathcal{L}(\lambda) &= \mathbb{E}_{q(x_{1:T}\,;\lambda)}\left[\log p(x_{1:T}, y_{1:T}) - \log q(x_{1:T}\,;\lambda)\right]\\
&= -\int\left\{p(x_{1:T}^{b_{1:T}}, y_{1:T})\mathbb{E}_{\widetilde{\phi}(x_{1:T}^{\neg b_{1:T}}, a_{1:T-1}^{\neg b_{1:T-1}}\,;\lambda)}\left[\frac{1}{\prod_{t=1}^T\frac{1}{N}\sum_{i=1}^N w_t^i}\right]\cdot\right.\\
&\qquad\left.\cdot\log\mathbb{E}_{\widetilde{\phi}(x_{1:T}^{\neg b_{1:T}}, a_{1:T-1}^{\neg b_{1:T-1}}\,;\lambda)}\left[\frac{1}{\prod_{t=1}^T\frac{1}{N}\sum_{i=1}^N w_t^i}\right]\right\}\mathrm{d}x_{1:T}^{b_{1:T}}.
\end{aligned} \tag{15}$$

Note that $-t \log t$ is a concave function for $t > 0$, this means by the conditional Jensen's inequality we have $-\mathbb{E}[t] \log \mathbb{E}[t] \geq -\mathbb{E}[t \log t]$. If we apply this to (15) we get

$$\begin{aligned}\mathcal{L}(\lambda) &\geq \int \mathbb{E}_{\widetilde{\phi}(x_{1:T}^{\neg b_{1:T}}, a_{1:T-1}^{\neg b_{1:T-1}}; \lambda)} \left[\frac{p(x_{1:T}^{b_{1:T}}, y_{1:T})}{\prod_{t=1}^{T} \frac{1}{N} \sum_{i=1}^{N} w_t^i} \sum_{t=1}^{T} \log \left(\frac{1}{N} \sum_{i=1}^{N} w_t^i \right) \right] \mathrm{d}x_{1:T}^{b_{1:T}} \\ &= \mathbb{E}_{\widetilde{\phi}(x_{1:T}^{1:N}, a_{1:T-1}^{1:N}; \lambda)} \left[\sum_{t=1}^{T} \log \left(\frac{1}{N} \sum_{i=1}^{N} w_t^i \right) \right] = \widetilde{\mathcal{L}}(\lambda),\end{aligned}$$

where the last step follows because $q(x_{1:T}; \lambda)$ is the marginal of $\widetilde{\phi}(x_{1:T}^{1:N}, a_{1:T-1}^{1:N}; \lambda)$. □

A.3 Stochastic Optimization

For the control variates we use

$$\sum_{t=2}^{T} c_t \mathbb{E}_{s(\cdot)\widetilde{\phi}(\cdot|\cdot;\lambda)} \left[\sum_{i=1}^{N} \nabla \log w_{t-1}^{a_{t-1}^i} - \sum_{\ell=1}^{N} \frac{w_{t-1}^{\ell}}{\sum_m w_{t-1}^m} \nabla \log w_{t-1}^{\ell} \right]$$

where

$$c_t = \mathbb{E}_{s(\cdot)\widetilde{\phi}(\cdot|\cdot;\lambda)} \left[\sum_{t'=t}^{T} \log \left(\frac{1}{N} \sum_{i=1}^{N} w_{t'}^i \right) \right].$$

In practice we use a stochastic estimate of c_t.

For $T = 2$ we can use a leave-one-out estimator of the ancestor variable score function gradient

$$\sum_{i=1}^{N} \mathbb{E}_{s(\cdot)\widetilde{\phi}(\cdot|\cdot;\lambda)} \left[\log \left(\frac{N-1}{N} \frac{\sum_{\ell=1}^{N} w_2^{\ell}}{\sum_{j \neq i} w_2^j} \right) \left(\nabla \log w_1^{a_1^i} - \sum_{\ell=1}^{N} \frac{w_1^{\ell}}{\sum_m w_1^m} \nabla \log w_1^{\ell} \right) \right].$$

Score Function Gradient Below we provide the derivation of a score function-like estimator that is applicable in very general settings. However, we have found that in practice the variance tends to be quite high.

$$\begin{aligned}\nabla \widetilde{\mathcal{L}}(\lambda) &= \nabla \mathbb{E}_{\widetilde{\phi}(x_{1:T}^{1:N}, a_{1:T-1}^{1:N}; \lambda)} \left[\log \widehat{p}(y_{1:T}) \right] \\ &= \mathbb{E}_{\widetilde{\phi}(x_{1:T}^{1:N}, a_{1:T-1}^{1:N}; \lambda)} \left[\nabla \log \widehat{p}(y_{1:T}) + \log \widehat{p}(y_{1:T}) \nabla \log \widetilde{\phi}(x_{1:T}^{1:N}, a_{1:T-1}^{1:N}; \lambda) \right],\end{aligned}$$

with

$$\nabla \log \widehat{p}(y_{1:T}) = \nabla \sum_{t=1}^{T} \log \left(\frac{1}{N} \sum_{i=1}^{N} w_t^i \right) = \sum_{t=1}^{T} \sum_{i=1}^{N} \frac{w_t^i}{\sum_{\ell} w_t^{\ell}} \nabla \log w_t^i,$$

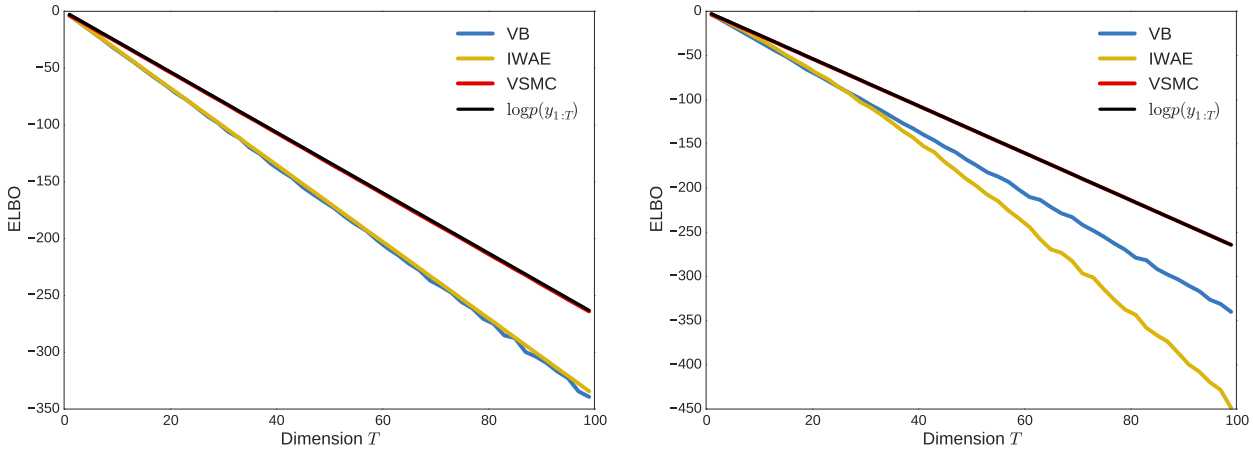
and

$$\begin{aligned}&\nabla \log \widetilde{\phi}(x_{1:T}^{1:N}, a_{1:T-1}^{1:N}; \lambda) \\ &= \sum_{i=1}^{N} \left[\nabla \log r(x_1^i; \lambda) + \sum_{t=2}^{T} \left[\nabla \log r(x_t^i | x_{t-1}^{a_{t-1}^i}; \lambda) + \nabla \log w_{t-1}^{a_{t-1}^i} - \sum_{\ell=1}^{N} \bar{w}_{t-1}^{\ell} \nabla \log w_{t-1}^{\ell} \right] \right].\end{aligned}$$

A.4 Scaling With Dimension

In this section we study how the methods compare on a simple toy model defined by

$$p(x_{1:T}, y_{1:T}) = \prod_{t=1}^{T} \mathcal{N}(x_t; 0, 1) \mathcal{N}(y_t; x_t^2, 1).$$



(a) $\lambda_{\text{IWAE}} = \lambda^{\star}_{\text{VB}}$ (b) $\lambda_{\text{IWAE}} = \lambda^{\star}_{\text{VSMC}}$

Figure 6: ELBO, for standard VB, IWAE, and VSMC, as a function of the dimension T of a toy problem. Here we set the number of samples in IWAE and VSMC to be $N = 2T$.

We study the data set $y_t = 3, \forall t$. Figure 6 shows the result when we let the number of samples in importance weighted auto-encoder (IWAE) (variational importance sampling (VIS)) and VSMC grow with the dimension $N = 2T$. For low T the optimal parameters for IWAE are close to $\lambda^{\star}_{\text{VSMC}}$. On the other hand for high T, the optimal parameters for IWAE are close to those of standard variational Bayes (VB), i.e. $\lambda^{\star}_{\text{VB}}$. Figure 6 indicates that just by letting $N \propto T$, VSMC can achieve arbitrarily good approximation of $p(x_{1:T} \mid y_{1:T})$ even if $T \to \infty$. This holds, under some regularity conditions, even if $p(x_{1:T}, y_{1:T})$ is a state space model [Bérard et al., 2014]. This asymptotic approximation property is not satisfied by VIS, we see in Figure 6 that the approximation deteriorates as T increases. Note that this does not hold if the dimension of the latent space, i.e. $\dim(x_t)$, tends to infinity rather than the number of time points T.