Variational Objectives for Markovian Dynamics with Backward Simulation

Antonio Khalil Moretti[*] [1] and **Zizhao Wang**[*] and **Luhuan Wu**[*] and **Iddo Drori** and **Itsik Pe'er**

Columbia University

Abstract. Sequential Monte Carlo (SMC) and Variational Inference (VI) are two families of approximate inference algorithms for Bayesian latent variable models. A body of recent work has focused on constructing a variational family of filtered distributions using SMC. Inspired by this work, we introduce Particle Smoothing Variational Objectives (SVO), a novel backward simulation technique and variational objective constructed from a smoothed approximate posterior. Our method sub-samples auxiliary random variables to enhance the support of the proposal distribution and increase particle diversity. We demonstrate our approach on three benchmark latent nonlinear dynamical systems tasks. SVO consistently outperforms filtered objectives when given fewer Monte Carlo samples.

1 Introduction

Latent variable models for time series are often formalized as a set of ordered, discrete-time measurements taken on a hidden dynamical system. A collection of recent work is concerned with inferring both the latent trajectories and latent dynamics of these systems when transition and emission functions are nonlinear [2, 5, 11, 16, 23, 24, 27]. Variational Inference (VI) and Sequential Monte Carlo (SMC) are two families of approximate inference algorithms for non-linear or non-conjugate Bayesian models. Recently, connections have been established between VI and SMC by using the latter to define a flexible variational family for hidden Markov models [19, 21, 25].

Standard variational SMC methods construct a *filtered* estimate of the log marginal likelihood which is used to specify a variational objective by forming a lower bound to the evidence [19, 21, 25, 26, 29]. This enables model learning and inference at the same time. In this approach, however, both the state-sequence and the objective are estimated using information only up to the current time point. This results in degraded posterior estimations when there exists significant observation noise or the system is partially observable. In contrast, particle smoothing methods generate a state-sequence conditioned on future observations [1, 3, 8, 15, 28]. This leads to improved inferred trajectories when the hidden dynamical system is described by a highly nonlinear or chaotic differential equation [11, 27]. For example, neurobiologists measuring a single-dimensional voltage trace are often interested in recovering nonlinear latent dynamics and trajectories that can be characterized using systems of coupled differential equations such as the Hodgkin Huxley [12]. However, two limitations of the existing particle smoothing literature are as follows:

i) Learning the model parameters that define the transition and emission functions is a distinct task typically handled using an EM algorithm.

ii) The majority of particle smoothing methods do not directly provide an unbiased estimate of the marginal likelihood [3, 15], thus making the construction of a smoothing-based variational objective a challenge.

We highlight the contributions of this paper as follows:

- **Particle Smoothing Variational Objective**: We propose Smoothing Variational Objectives (SVO), a framework for performing VI on nonlinear hidden Markov models. SVO jointly estimates the model parameters and the marginal likelihood from the smoothed state-sequence, analogous to the approach of the variational auto-encoder. SVO is a novel recursive backward-sampling algorithm and approximate smoothing posterior defined through a subsampling process. This augments the support of the proposal and boosts particle diversity.
- **Unbiased Likelihood Estimator**: We prove that SVO generates an unbiased estimate of the marginal likelihood from the backward state-sequence. We explore the ability of SVO to recover nonlinear embeddings, transition and emission functions from only the observations. To quantify the learned dynamics, we repeatedly apply the trained transition function in the target to propagate the system forwards without input data and then use the emission function to make observation predictions. We show that our smoothed objective generates an improved estimate of the latent state as measured by the ability of the target to more accurately predict observations using the dynamics learned.
- **Applications**: We demonstrate our approach on to three benchmark latent nonlinear dynamical systems tasks, including single cell voltage trace data. SVO outperforms filtered objectives when given fewer Monte Carlo samples on all three tasks.

2 Preliminaries

Inference in State Space Models Let $\mathbf{X} \equiv \{\mathbf{x}_1, \dots \mathbf{x}_T\}$ denote a sequence of T observations of a $\mathbb{R}^{d_{\mathbf{x}}}$-dependent random variable. State space models (SSMs) posit a generating process for $\mathbf{X}$ through a sequence $\mathbf{Z} \equiv \{\mathbf{z}_1, \dots \mathbf{z}_T\}$, $\mathbf{z}_t \in \mathbb{R}^{d_{\mathbf{z}}}$ of unobserved latent variables, that transitions according to a stochastic evolution law. The

[1] Columbia University, Department of Computer Science, USA, email: amoretti@cs.columbia.edu. $*$ denotes equal contribution.

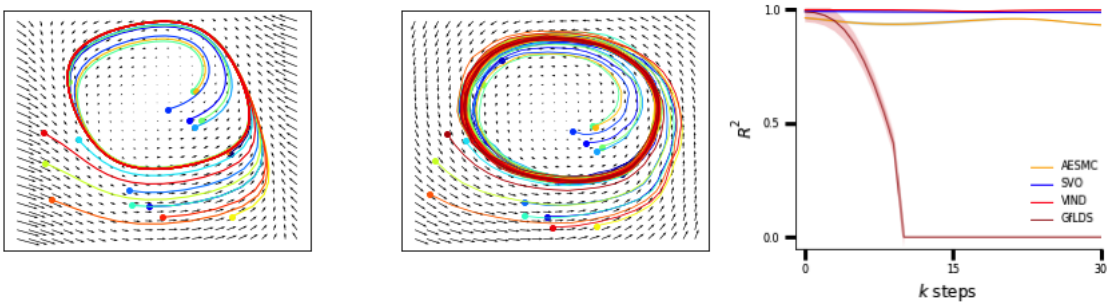

Figure 1. Summary of the Fitzhugh-Nagumo results: the observation is one-dimensonal while the phase space and latent variables are two-dimensional; (left) ground truth dynamics and trajectories for the original system; (center) latent dynamics and trajectories inferred by SVO; Initial points (denoted by markers) located both inside and outside the limit cycle are topologically invariant in the SVO reconstruction; (right) R_k^2 for various models on the dimensionality expansion task. Results are averaged over 3 random seeds.

joint density then factorizes:

$$p_\theta(\mathbf{X}, \mathbf{Z}) = F_\theta(\mathbf{Z}) \cdot \prod_{t=1}^{T} g_\theta(\mathbf{x}_t|\mathbf{z}_t)\,, \tag{1}$$

where $g_\theta(\mathbf{x}|\mathbf{z})$ is an observation model, and $F_\theta(\mathbf{Z})$ is a prior representing the evolution in the latent space. In this work, we focus on the case of Markov evolution with Gaussian conditionals:

$$F_\theta(\mathbf{Z}) = f_1(\mathbf{z}_1) \prod_{t=2}^{T} f_\theta(\mathbf{z}_t|\mathbf{z}_{t-1})\,,$$

$$f_1 = \mathcal{N}\big(\psi_1, \mathbf{Q}_1\big)\,, \qquad \mathbf{z}_t \sim \mathcal{N}\big(\psi_\theta(\mathbf{z}_{t-1}),\, \mathbf{Q}\big)\,. \tag{2}$$

Inference in SSMs requires marginalizing the joint distribution with respect to the hidden variables $\mathbf{Z}$,

$$\log p_\theta(\mathbf{X}) = \int \log p_\theta(\mathbf{X}, \mathbf{Z})\, d\mathbf{Z}. \tag{3}$$

This procedure is intractable when $\psi_\theta(\mathbf{z}_t)$ is a nonlinear function or when $g_\theta(\mathbf{x}_t|\mathbf{z}_t)$ is non-Gaussian.

Variational Inference VI describes a family of techniques for approximating $\log p_\theta(\mathbf{X})$ when marginalization is analytically impossible. The idea is to define a tractable distribution $q_\phi(\mathbf{Z}|\mathbf{X})$ and then optimize a lower bound to the log-likelihood:

$$\log p_\theta(\mathbf{X}) \geq \mathcal{L}_{\text{ELBO}}(\theta, \phi, \mathbf{X}) = \mathop{\mathbb{E}}_{q}\left[\log \frac{p_\theta(\mathbf{X}, \mathbf{Z})}{q_\phi(\mathbf{Z}|\mathbf{X})}\right]. \tag{4}$$

Tractability and expressiveness of the variational approximation $q_\phi(\mathbf{Z}|\mathbf{X})$ are contrasting goals. Auto Encoding Variational Bayes [14] (AEVB) is a method to simultaneously train $q_\phi(\mathbf{Z}|\mathbf{X})$ and $p_\theta(\mathbf{X}, \mathbf{Z})$. The expectation value in Eq. (4) is approximated by summing over samples from the recognition distribution; which in turn are drawn by evaluating a deterministic function of a ϕ-independent random variable (the reparameterization trick). Building upon this, the Importance Weighted Auto Encoder [4, 6] (IWAE) constructs tighter bounds than the AEVB through mode averaging as opposed to mode matching. The idea to achieve a better estimate of the log-likelihood is to draw K samples from the proposal and to average probability ratios.

Filtering SMC SMC is a family of techniques for inference in SSMs with an intractable joint. Given a proposal distribution $q_\phi(\mathbf{Z}|\mathbf{X})$, these methods operate sequentially, approximating $p_\theta(\mathbf{z}_{1:t}|\mathbf{x}_{1:t})$ (the *target*) for each t by performing inference on a sequence of increasing probability spaces. K samples (*particles*) are drawn from a proposal distribution and used to compute importance weights:

$$\mathbf{z}_t^k \sim q_\phi(\mathbf{z}_t^k|\mathbf{z}_{t-1}^k, \mathbf{x}_t)\,, \quad w_t^k \coloneqq \frac{f_\theta(\mathbf{z}_t^k|\mathbf{z}_{t-1}^k) g_\theta(\mathbf{x}_t|\mathbf{z}_t^k)}{q_\phi(\mathbf{z}_t^k|\mathbf{z}_{t-1}^k, \mathbf{x}_t)}. \tag{5}$$

A resampling strategy ensures that particles remain on regions of high probability mass. SMC accomplishes this goal by resampling the particle indices (*ancestors*) according to their weights at the previous time step:

$$a_{t-1}^k \sim \textsc{Categorical}(\cdot|\bar{w}_{t-1}^1, \cdots, \bar{w}_{t-1}^K)\,, \tag{6}$$
$$w_t^k \coloneqq \frac{f_\theta(\mathbf{z}_t^k|\mathbf{z}_{t-1}^{a_{t-1}^k}) g_\theta(\mathbf{x}_t|\mathbf{z}_t^{a_{t-1}^k})}{q_\phi(\mathbf{z}_t^k|\mathbf{z}_{t-1}^{a_{t-1}^k}, \mathbf{x}_t)}.$$

The posterior can be evaluated at the final time step. The functional integral is approximated below where $\delta_{\mathbf{z}_{1:T}^k}(\mathbf{z}_{1:T})$ is the Dirac measure:

$$\sum_{k=1}^{K} \bar{w}_T^k \delta_{\mathbf{z}_{1:T}^k}(\mathbf{z}_{1:T}) \quad \text{where} \quad \bar{w}_T^k = w_T^k / \sum_{j=1}^{K} w_T^j. \tag{7}$$

The SMC algorithm is deterministic conditioning on $(\mathbf{z}_{1:T}^{1:K}, a_{1:T-1}^{1:K})$ [21, 19]. This implies that the proposal density can be reparameterized to act as a variational distribution that can be encoded:

$$Q_{\text{SMC}}(\mathbf{Z}_{1:T}^{1:K}, \mathbf{A}_{1:T-1}^{1:K}) \coloneqq \left(\prod_{k=1}^{K} q_{1,\phi}(\mathbf{z}_1^k)\right) \tag{8}$$
$$\times \prod_{t=2}^{T} \prod_{k=1}^{K} q_{t,\phi}(\mathbf{z}_t^k|\mathbf{z}_{1:t-1}^{a_{t-1}^k}) \cdot \textsc{Categorical}(a_{t-1}^k|\bar{w}_{t-1}^{1:K}).$$

An unbiased estimate for the marginal likelihood and the corresponding objective are defined below:

$$\hat{\mathcal{Z}}_{\text{SMC}} \coloneqq \prod_{t=1}^{T} \Big[\frac{1}{K} \sum_{k=1}^{K} w_t^k\Big], \quad \mathcal{L}_{\text{SMC}} \coloneqq \mathop{\mathbb{E}}_{Q_{\text{SMC}}}\Big[\log \hat{Z}_{\text{SMC}}\Big]. \tag{9}$$

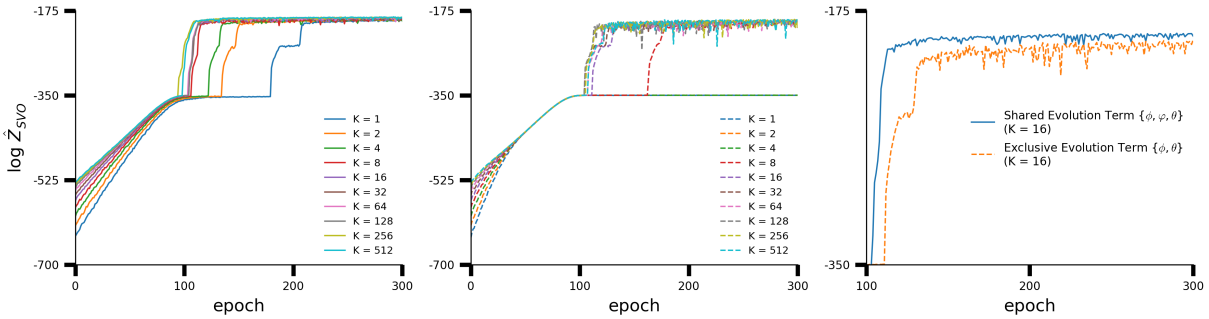

Figure 2. ELBO convergence across epochs for SVO using exclusive parameters θ, ϕ and shared parameters θ, φ, ϕ; (left) log $\hat{\mathcal{Z}}_{SVO}$ across epochs as K increases using shared evolution network; (center) log $\hat{\mathcal{Z}}_{SVO}$ across epochs as K increases using independent evolution networks; (right) log $\hat{\mathcal{Z}}_{SVO}$ convergence for shared vs independent evolution networks with $K = 16$ highlighting faster convergence to a higher ELBO.

Particle Smoothing with Backward Simulation Forward Filtering Backward Simulation (FFBSi) [8] is an approach to approximate the smoothing posterior which admits the following factorization

$$p(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) = p(\mathbf{z}_T|\mathbf{x}_{1:T}) \prod_{t=1}^{T-1} p(\mathbf{z}_t|\mathbf{z}_{t+1:T}, \mathbf{x}_{1:T}) \,, \tag{10}$$

where, by Markovian assumptions, the conditional backward kernel can be written as:

$$p(\mathbf{z}_t|\mathbf{z}_{t+1}, \mathbf{x}_{1:T}) \propto p(\mathbf{z}_t|\mathbf{x}_{1:t}) f(\mathbf{z}_{t+1}|\mathbf{z}_t). \tag{11}$$

FFBSi begins with filtering to obtain $\{\mathbf{z}_{1:T}^{1:K}, w_{1:T}^{1:K}\}$ which provides a particulate approximation to the backward kernel:

$$p(\mathbf{z}_t|\mathbf{z}_{t+1}, \mathbf{x}_{1:T}) \approx \sum_{i=1}^{K} w_{t|t+1}^{k} \delta_{\mathbf{z}_t^i}(\mathbf{z}_t), \tag{12}$$

$$\text{where} \qquad w_{t|t+1}^{i} = \frac{w_t^i f(\mathbf{z}_{t+1}|\mathbf{z}_t^i)}{\sum_{j=1}^{K} w_t^j f(\mathbf{z}_{t+1}|\mathbf{z}_t^j)}.$$

Backward simulation generates states in the reverse-time direction conditioning on future states by choosing $\tilde{\mathbf{z}}_t = \mathbf{z}_t^i$ with probability $w_{t|T}^i$. This corresponds to a *discrete* resampling step in the backward pass. As a result the backward kernel is approximated from particles that are drawn from the proposal $q(\mathbf{z}_t|\mathbf{z}_{t-1})$ in the forward pass. The FFBSi can only generate trajectories suppported by the forward filtering particles, thus limiting the expressiveness of the variational distribution.

3 Related Work

AESMC [19], FIVO [21] and VSMC [25] are three closely related methods that form a lower bound to the log marginal likelihood which is estimated using filtering SMC, however without conditioning the latent state on future observations they may fail to capture long-term dependencies. VSMC draws a single sample at the final time step to produce a trajectory from the corresponding ancestral path. While this heuristic produces *one* sample conditioned on all observations, the resulting path is not used to construct the surrogate ELBO which is filtered.

Particle Smoothing Particle smoothing methods include the previously discussed FFBSi [8] and the Two Filter Smoother (TFS) [28]. The FFBSi defines a posterior over an entire trajectory and gives a way to sample the trajectory backward in time. In contrast, TFS defines a posterior only at a single time step. Additionally they differ in their methods. For TFS, the backward filtering is independent of the forward filtering. However, our backward simulation is conditional on forward filtering, where the subweight depends on the forward system. Unlike standard particle smoothing methods, SVO is a framework for performing VI on state-space models, jointly for the states and the model itself, analogous to the approach of the variational auto-encoder [14]. The proposal and the target distribution are trained from the observation sequence.

Computational Complexity Particle smoothing methods incur a cost that is quadratic in the number of particles due to the pairwise interactions to be defined in Eq (13). For SVO, smoothing incurs a cost of $\mathcal{O}(TK^2Md_z)$ operations in contrast to $\mathcal{O}(TGd_z)$ in AESMC (where G denotes the number of particles). For a fair comparison in the experiments to follow we give AESMC the corresponding extra particles. Empirically, SVO with small K and M (4 or 8) can provide a more accurate posterior approximation than AESMC with a much larger value of G. For the FHN task, SVO with $K = M = 32$ outperforms AESMC with $G = 1024$ (see Fig 1); For the Lorenz task SVO with $K = M = 2$ also outperforms AESMC with $G = 256$ (see Fig 3). SVO also works with larger T, K, and M (with $T = 1000$ on the Allen data). All the experiments were run on 16 core CPU machines. Despite the $\mathcal{O}(TK^2M)$ complexity, the main cost is evaluating the neural network $\psi(\cdot)$ and its gradients for $f(\cdot|\mathbf{z}_{t-1}^j) = \mathcal{N}(\cdot|\psi(\mathbf{z}_{t-1}^j), \Sigma)$. The computation is $\mathcal{O}(TK)$ here and $\mathcal{O}(TKM)$ in the emission term. The $\mathcal{O}(TK^2M)$ is fast relative to the evaluation of the neural network.

Variational Methods Two variational smoothing methods for inference in non-conjugate SSMs are GfLDS [7, 2] and VIND [11]. These methods simultaneously train generative models and variational approximations analogous to proposal and target distributions in SVO. GfLDS is a generative model and approximation for linear latent dynamics together with nonlinear emission densities. Building upon this, VIND is governed by nonlinear latent dynamics and emissions. GfLDS and VIND both require inverting a block-tridiagonal matrix which mixes components of state space through the inverse covariance. This incurs a complexity of $\mathcal{O}(Td_z^3)$ where T is the length of the time series and d_z is the state dimension. An alternative approach is to directly modify the target distribution in SMC to achieve smoothing [10]. TVSMC [18] and SMC-Twist [20] augment the intermediate target distribution with a twisting function, which in turn is approximated with deterministic algorithms such as tem-

poral difference learning and Laplace approximation. When applied to nonlinear time series it was reported that TVSMC underperforms relative to filtering using VSMC [18].

4 Particle Smoothing Variational Objectives

We will utilize the smoothing posterior in Eq. (10) to define a backward proposal distribution and sample trajectories to construct a variational objective. We propose a novel approximate posterior to overcome the limitation of the FFBSi by augmenting the support of the backward kernel through the subsampling of auxiliary random variables.

Overview We provide an overview of Particle Smoothing Variational Objectives (SVO) before presenting a detailed derivation and description in Algorithm 1 (we have annotated the overview with steps from the algorithm). Smoothing is based on filtering SMC which provides the forward weights and particles $\{\mathbf{z}_{1:T}^{1:K}, w_{1:T}^{1:K}\}$ (*step 1*). With outputs from filtering SMC, SVO proceeds to generate backward trajectories. This is done by approximating a sequence of backward posteriors through a process of self-normalized importance sampling. At time T, for each trajectory we will draw M *subparticles* from a continuous-domain conditional kernel (*step 3*). While the final time step requires some care, these subparticles will be used to initialize *subweights* relative to the conditional kernel (*step 4*). The subweights in turn, are used to update the corresponding particle by drawing a backward index from a resampling process (*step 5*). The trajectory is initialized with the selected particle and extended sequentially (*step 6*). SVO iterates by drawing M subparticles from a continuous-domain backward proposal for each of the K trajectories at the current time step (*step 9*). SVO then computes subweights for each subparticle (*step 10*) in order to select a single backward particle from the set of M candidates (*step 11*). Finally the backward kernel is evaluated using the chosen resampled particle (*step 13*). The output of this procedure is a collection of particle trajectories from the smoothing posterior that are used to define a variational objective.

Objective Function We introduce a *continuous* reverse-dynamics proposal $q(\mathbf{z}_t|\mathbf{z}_{t+1}, \mathbf{x}_{1:T})$ that is used to sample M subparticles for each $k \in \{1, \cdots, K\}$, $\tilde{\mathbf{z}}_t^{k,1:M} \sim q(\mathbf{z}_t|\tilde{\mathbf{z}}_{t+1}^k, \mathbf{x}_{1:T})$. These samples are used to define subweights as follows

$$
\begin{aligned}
&p(\tilde{\mathbf{z}}_t^{k,m}|\tilde{\mathbf{z}}_{t+1}^{k,m}, \mathbf{x}_{1:T})/q(\tilde{\mathbf{z}}_t^{k,m}|\tilde{\mathbf{z}}_{t+1}^{k}, \mathbf{x}_{1:T}) \\
&\propto \int p(\mathbf{z}_{t-1}, \tilde{\mathbf{z}}_t^{k,m}|\mathbf{x}_{1:t-1}) d\mathbf{z}_{t-1} \frac{f(\tilde{\mathbf{z}}_{t+1}^k|\tilde{\mathbf{z}}_t^{k,m}) g(\mathbf{x}_t|\tilde{\mathbf{z}}_t^{k,m})}{q(\tilde{\mathbf{z}}_t^{k,m}|\tilde{\mathbf{z}}_{t+1}^k, \mathbf{x}_{1:T})} \\
&\approx \left[\sum_{j=1}^K \bar{w}_{t-1}^j f(\tilde{\mathbf{z}}_t^{k,m}|\mathbf{z}_{t-1}^j)\right] \frac{f(\tilde{\mathbf{z}}_{t+1}^k|\tilde{\mathbf{z}}_t^{k,m}) g(\mathbf{x}_t|\tilde{\mathbf{z}}_t^{k,m})}{q(\tilde{\mathbf{z}}_t^{k,m}|\tilde{\mathbf{z}}_{t+1}^k, \mathbf{x}_{1:T})} \\
&\coloneqq \omega_{t|T}^{k,m}.
\end{aligned} \tag{13}
$$

A single particle is selected by sampling an index with probability proportional to the subweight $\omega_{t|T}$: $b_t^k \sim \text{CATEGORICAL}(b_t^k|\omega_{t|T}^{k,1}, \cdots, \omega_{t|T}^{k,M})$, $\tilde{\mathbf{z}}_t^k \leftarrow \tilde{\mathbf{z}}_t^{k,b_t^k}$. This modified particulate distribution now generates hidden states from a *continuous* domain given the future state and all observations. Repeating this process sequentially in the reverse-time direction produces K i.i.d. sample trajectories, $\{\tilde{\mathbf{z}}_{1:T}^{1:K}\}$ (see Algorithm 1).

The approximate posterior and variational objective are defined below via Algorithm 1. Note again that the following expectations are also conditioned on the forward filtering system.

Algorithm 1: Particle Smoothing Variational Objectives

1. Perform forward filtering to obtain $\{\mathbf{z}_{1:T}^{1:K}, w_{1:T}^{1:K}\}$
2. Initialization. For $k = 1, \cdots, K$:
   3. Sample M subparticles: $\{\tilde{\mathbf{z}}_T^{k,m}\}_{m=1}^M \sim q(\cdot|\mathbf{x}_{1:T})$
   4. Initialize subweight for each subparticle:
   $$\omega_{T|T}^{k,m} \propto \Big[\sum_j \bar{w}_{T-1}^j f(\tilde{\mathbf{z}}_T^{k,m}|\mathbf{z}_{T-1}^j)\Big] \frac{g(\mathbf{x}_T|\tilde{\mathbf{z}}_T^{m,k})}{q(\tilde{\mathbf{z}}_T^{k,m}|\mathbf{x}_{1:T})}$$
   5. Sample index: $b_T^k \sim \text{CATEGORICAL}(\cdot|\omega_{T|T}^{k,1}, \cdots, \omega_{T|T}^{k,M})$
   6. Set backward particle: $\tilde{\mathbf{z}}_T^k \leftarrow \tilde{\mathbf{z}}_T^{k,b_t^m}$, $\omega_{T|T}^k \leftarrow \omega_{T|T}^{k,b_T^k}$
   7. Evaluate the backward proposal: $\Omega_T^k \coloneqq M \cdot \omega_{T|T}^k \cdot q(\tilde{\mathbf{z}}_T^k|\mathbf{x}_{1:T})$,
8. Backward Simulation. For $t = T-1, \cdots, 1$ and $k = 1, \cdots, K$:
   9. Sample M subparticles from reverse-dynamics proposal: $\{\tilde{\mathbf{z}}_t^{k,m}\}_{m=1}^M \sim q(\cdot|\tilde{\mathbf{z}}_{t+1}^k, \mathbf{x}_{1:T})$
   10. Compute subweights:
   $$\omega_{t|T}^{k,m} \propto \sum_j \bar{w}_{t-1}^j f(\tilde{\mathbf{z}}_t^{k,m}|\mathbf{z}_{t-1}^j) \times \frac{f(\tilde{\mathbf{z}}_{t+1}^k|\tilde{\mathbf{z}}_t^{k,m}) g(\mathbf{x}_t|\tilde{\mathbf{z}}_t^{k,m})}{q(\tilde{\mathbf{z}}_t^{k,m}|\tilde{\mathbf{z}}_{t+1}^k, \mathbf{x}_{1:T})}$$
   11. Sample index. $b_t^k \sim \text{CATEGORICAL}(\cdot|\omega_{t|T}^{k,1}, \cdots, \omega_{t|T}^{k,M})$
   12. Set backward particle: $\tilde{\mathbf{z}}_t^k \leftarrow \tilde{\mathbf{z}}_t^{k,b_t^m}$, $\omega_{t|T}^k \leftarrow \omega_{t|T}^{k,b_t^k}$
   13. Evaluate the backward proposal: $\Omega_t^k = M \cdot \omega_{t|T}^k \cdot q(\tilde{\mathbf{z}}_t^k|\tilde{\mathbf{z}}_{t+1}^k, \mathbf{x}_{1:T})$
14. **return**

$$\tilde{\mathbf{z}}_{1:T}^{1:K}, \hat{\mathcal{L}}_{SVO}(\mathbf{x}_{1:T}) \coloneqq \log\left(\frac{1}{K}\sum_{k=1}^K \frac{p(\tilde{\mathbf{z}}_{1:T}^k, \mathbf{x}_{1:T})}{\prod_{t=1}^T \Omega_t^k}\right)$$

$$\mathcal{L}_{SVO} \coloneqq \mathop{\mathbb{E}}_{q}\Big[\log \hat{\mathcal{Z}}_{SVO}\Big], \quad \hat{\mathcal{Z}}_{SVO} \coloneqq \frac{1}{K}\sum_{k=1}^K \frac{p(\tilde{\mathbf{z}}_{1:T}^k, \mathbf{x}_{1:T})}{q(\tilde{\mathbf{z}}_{1:T}^k|\mathbf{x}_{1:T})}, \tag{14}$$

where $q(\tilde{\mathbf{z}}_{1:T}^k|\mathbf{x}_{1:T}) \coloneqq$

$$M^T \cdot \omega_{T|T}^k \cdot q(\tilde{\mathbf{z}}_T^k|\mathbf{x}_{1:T}) \prod_{t=1}^{T-1}\Big[\omega_{t|T}^k \cdot q(\tilde{\mathbf{z}}_t^k|\tilde{\mathbf{z}}_{t+1}^k, \mathbf{x}_{1:T})\Big]. \tag{15}$$

We note that while the sequence of target distributions is filtered, our objective is constructed using samples from a smoothing posterior. This heuristic facilitates smoothing the target when performing VI to simultaneously train $p(\mathbf{Z}|\mathbf{X})$ and $q(\mathbf{Z}|\mathbf{X})$ by pulling $p(\mathbf{Z}|\mathbf{X}) \rightarrow q(\mathbf{Z}|\mathbf{X})$. This functional dependence motivates sharing the transition function between proposal and target.

Theorem 1. *$\hat{Z}_{SVO}$ is an unbiased estimate of $p(\mathbf{x}_{1:T})$.*

$$\mathbb{E}_{Q(\hat{\mathbf{z}}_{1:T}^{1:K,1:M})}\left[\frac{1}{K}\sum_{k=1}^K \frac{p(\hat{\mathbf{z}}_{1:T}^k, \mathbf{x}_{1:T})}{\prod_{t=1}^T \Omega_t^K}\right] = p(\mathbf{x}_{1:T}),$$

where $Q(\hat{\mathbf{z}}_{1:T}^{1:K,1:M})$ denotes the sampling distribution of $\hat{z}_{1:T}^{1:M}$ according to Algorithm 1.

Proof. We will define auxiliary variables λ and distributions $q(\lambda|x)$, $q(z|\lambda, x)$, and $r(\lambda|z, x)$ such that

$$\hat{\mathcal{Z}}_{SVO} \equiv \hat{p}(x) = \frac{p(x,z)r(\lambda|z,x)}{q(z,\lambda|x)} = \frac{p(x,z)r(\lambda|z,x)}{q(z|\lambda,x)q(\lambda|x)},$$

where $z, \lambda \sim q(z, \lambda|x)$. For a treatment of auxiliary random variables see [6, 17]. Here the auxiliary latent variables are the unselected subparticles,

$$\lambda = \{\tilde{\mathbf{z}}_{1:T}^{\neg b_{1:T}^{1:K}}\}.$$

For convenience, we omit the conditioning on the forward system. To further simplify notation, we will rearrange particles to omit the backward ancestor indices by defining $\hat{\mathbf{z}}_t^{k,1} \leftarrow \tilde{\mathbf{z}}_t^{k,b_t^k}$, $\hat{\omega}_{t|T}^k \leftarrow \omega_{t|T}^{k,b_t^k}$ and $\hat{\mathbf{z}}_t^{k,2:M} \leftarrow \tilde{\mathbf{z}}_t^{k,\neg b_t^k}$, $\hat{\omega}_{t|T}^{k,2:M} \leftarrow \omega_{t|T}^{k,\neg b_t^k}$. By the linearity of expectation, it suffices to show the case of $K = 1$ (as a result, for clarity, we will omit k, in the superscripts):

$$\mathbb{E}_{\hat{\mathbf{z}}_{1:T}^{1:M}}\left[\frac{p(\hat{\mathbf{z}}_{1:T}^1, \mathbf{x}_{1:T})}{\prod_{t=1}^T \Omega_t}\right] = p(\mathbf{x}_{1:T})$$

We begin by expressing the generative distribution of the sampling process for the rearranged particles $\hat{\mathbf{z}}_{1:T}^{1:M}$ as factorizing:

$$Q(\hat{\mathbf{z}}_{1:T}^{1:M}|\mathbf{x}_{1:T}) = Q(\hat{\mathbf{z}}_T^{1:M}|\mathbf{x}_{1:T}) \prod_{t=1}^{T-1} Q(\hat{\mathbf{z}}_t^{1:M}|\hat{\mathbf{z}}_{t+1}^1, \mathbf{x}_{1:T}).$$

Consider the sampling process last time step,

- Step 1. Sample $\{\tilde{\mathbf{z}}_T^m\}_{m=1}^M \sim q(\cdot|\mathbf{x}_{1:T})$, and compute the associated weights $\tilde{\omega}_{T|T}^{1:M}$ as outlined in Algorithm 1
- Step 2. Sample $b_T \sim \text{CATEGORICAL}(\cdot|\tilde{\omega}_{T|T}^1, \cdots, \tilde{\omega}_{T|T}^M)$
- Step 3. Set $\hat{\mathbf{z}}_T^1 \leftarrow \tilde{\mathbf{z}}_T^{b_T}$, $\hat{\mathbf{z}}_T^{2:M} \leftarrow \tilde{\mathbf{z}}_T^{\neg b_T}$, and $\hat{\omega}_{T|T}^1 \leftarrow \tilde{\omega}_{T|T}^{b_T}$, $\hat{\omega}_{T|T}^{2:M} \leftarrow \tilde{\omega}_{T|T}^{\neg b_T}$

The marginal distribution of $\hat{\mathbf{z}}_T^{1:M}$ is obtained as follows:

$$\begin{aligned}
&Q(\hat{\mathbf{z}}_T^{1:M}|\mathbf{x}_{1:T}) \\
&= \int \left[\prod_{m=1}^M \underbrace{q(\tilde{\mathbf{z}}_T^m|\mathbf{x}_{1:T})}_{\text{Step 1}}\right]\left[\sum_{b_T=1}^M \underbrace{\frac{p(b_T|\tilde{\mathbf{z}}_T^{1:M})}{\text{Step 2}}}\underbrace{p(\hat{\mathbf{z}}_T^{1:M}|\tilde{\mathbf{z}}_T^{1:M}, b_T)}_{\text{Step 3}}\right] d\tilde{\mathbf{z}}_T^{1:M} \\
&= \sum_{b_T=1}^M \int \left[\prod_{m=1}^M q(\tilde{\mathbf{z}}_T^m|\mathbf{x}_{1:T})\right] \cdot \frac{\tilde{\omega}_{T|T}^{b_T}}{\tilde{\omega}_{T|T}^{b_T} + \sum_{i\in\neg b_T}\tilde{\omega}^i}{}_{T|T} \\
&\qquad \cdot \delta(\hat{\mathbf{z}}_T^1 - \tilde{\mathbf{z}}_T^{b_T})\delta(\hat{\mathbf{z}}_T^{2:M} - \tilde{\mathbf{z}}_T^{\neg b_T}) d\tilde{\mathbf{z}}_T^{1:M}
\end{aligned}$$

Collapsing all possible cases to $b_T = 1$ by symmetry,

$$\begin{aligned}
&= M \int \left[\prod_{m=1}^M q(\tilde{\mathbf{z}}_T^m|\mathbf{x}_{1:T})\right] \cdot \frac{\tilde{\omega}_{T|T}^1}{\tilde{\omega}_{T|T}^1 + \sum_{i=2:M}\tilde{\omega}_{T|T}^i} \\
&\qquad \cdot \delta(\hat{\mathbf{z}}_T^1 - \tilde{\mathbf{z}}_T^1)\delta(\hat{\mathbf{z}}_T^{2:M} - \tilde{\mathbf{z}}_T^{2:M}) d\tilde{\mathbf{z}}_T^{1:M}
\end{aligned}$$

Integrating over the Dirac measures:

$$= M\left[\prod_{m=1}^M q(\hat{\mathbf{z}}_T^m|\mathbf{x}_{1:T})\right]\frac{\hat{\omega}_{T|T}^1}{\sum_{i=1}^M \hat{\omega}_{T|T}^i},$$

Similarly, we have the following for $t = 1, \ldots, T-1$,

$$Q(\hat{\mathbf{z}}_t^{1:M}|\hat{\mathbf{z}}_{t+1}^1, \mathbf{x}_{1:T}) = \left[\prod_{m=1}^M q(\hat{\mathbf{z}}_t^m|\hat{\mathbf{z}}_{t+1}^1, \mathbf{x}_{1:T})\right] \cdot M \cdot \frac{\hat{\omega}_{t|T}^1}{\sum_{m=1}^M \hat{\omega}_{t|T}^m}.$$

Therefore,

$$\begin{aligned}
&Q(\hat{\mathbf{z}}_{1:T}^{1:M}|\mathbf{x}_{1:T}) \\
&\qquad = \underbrace{\left[\prod_{t=1}^T \Omega_t\right]}_{q(z|\lambda,x)} \cdot \underbrace{\prod_{m=2}^M \left[q(\hat{\mathbf{z}}_T^m|\mathbf{x}_{1:T}) \prod_{t=1}^{T-1} q(\hat{\mathbf{z}}_t^m|\hat{\mathbf{z}}_{t+1}^1, \mathbf{x}_{1:T})\right]}_{q(\lambda|x)}.
\end{aligned}$$

Now, define the target distribution to be:

$$P(\hat{\mathbf{z}}_{1:T}^{1:M}, \mathbf{x}_{1:T}) = p(\hat{\mathbf{z}}_{1:T}^1, \mathbf{x}_{1:T}) r(\lambda|\mathbf{x}_{1:T}, \mathbf{z}_{1:T})$$

where

$$\begin{aligned}
r(\lambda|\mathbf{x}_{1:T}, \mathbf{z}_{1:T}^{1:M}) &= q(\lambda|\mathbf{x}_{1:T}) \\
&= \prod_{m=2}^M \left[q(\hat{\mathbf{z}}_T^m|\mathbf{x}_{1:T}) \prod_{t=1}^{T-1} q(\hat{\mathbf{z}}_t^m|\hat{\mathbf{z}}_{t+1}^1, \mathbf{x}_{1:T})\right].
\end{aligned}$$

Overall,

$$\mathbb{E}_{Q(\hat{\mathbf{z}}_{1:T}^{1:M})}[\hat{Z}_{SVO}] = \mathbb{E}_{Q(\hat{\mathbf{z}}_{1:T}^{1:M})}\left[\frac{p(\hat{\mathbf{z}}_{1:T}^1, \mathbf{x}_{1:T})}{\prod_{t=1}^T \Omega_t}\right]$$

Writing the augmented target and proposal explicitly:

$$\begin{aligned}
&= \mathop{\mathbb{E}}_{Q}\left[\frac{p(\hat{\mathbf{z}}_{1:T}^1, \mathbf{x}_{1:T}) \times \prod_{m=2}^M \left[q(\hat{\mathbf{z}}_T^m|\mathbf{x}_{1:T}) \prod_{t=1}^{T-1} q(\hat{\mathbf{z}}_t^m|\hat{\mathbf{z}}_{t+1}^1, \mathbf{x}_{1:T})\right]}{\prod_{t=1}^T \Omega_t \times \prod_{m=2}^M \left[q(\hat{\mathbf{z}}_T^m|\mathbf{x}_{1:T}) \prod_{t=1}^{T-1} q(\hat{\mathbf{z}}_t^m|\hat{\mathbf{z}}_{t+1}^1, \mathbf{x}_{1:T})\right]}\right] \\
&= \mathop{\mathbb{E}}_{Q}\left[\frac{P(\hat{\mathbf{z}}_{1:T}^{1:M}, \mathbf{x}_{1:T})}{Q(\hat{\mathbf{z}}_{1:T}^{1:M})}\right] \\
&= \int P(\hat{\mathbf{z}}_{1:T}^{1:M}, \mathbf{x}_{1:T}) d\hat{\mathbf{z}}_{1:T}^{1:M} \\
&= \int p(\hat{\mathbf{z}}_{1:T}^1, \mathbf{x}_{1:T}) \\
&\times \left[\int \prod_{m=2}^M \left[q(\hat{\mathbf{z}}_T^m|\mathbf{x}_{1:T}) \prod_{t=1}^{T-1} q(\hat{\mathbf{z}}_t^m|\hat{\mathbf{z}}_{t+1}^1, \mathbf{x}_{1:T})\right] d\hat{\mathbf{z}}_{1:T}^{2:M}\right] d\hat{\mathbf{z}}_{1:T}^1 \\
&= \int p(\hat{\mathbf{z}}_{1:T}^1, \mathbf{x}_{1:T}) d\hat{\mathbf{z}}_{1:T}^1 \\
&= p(\mathbf{x}_{1:T}).
\end{aligned}$$

□

Implementation Details

In the forward filtering pass, we define the proposal distribution as follows:

$$\begin{aligned}
q_{\phi,\varphi}(\mathbf{z}_{1:T}^k|\mathbf{x}_{1:T}) \propto \underbrace{f_\varphi(\mathbf{z}_1^k)}_{initial\ state} \prod_{t=1}^T \underbrace{h_\phi(\mathbf{z}_t^k|\mathbf{x}_t)}_{encoding} \\
\prod_{t=2}^T \underbrace{\text{CATEGORICAL}(a_{t-1}^k|\bar{w}_{t-1}^{1:K})}_{resampling} \underbrace{f_\varphi(\mathbf{z}_t^k|\mathbf{z}_{t-1}^{a_{t-1}^k})}_{transition},
\end{aligned} \tag{16}$$

where the proposal density factorizes into evolution and encoding functions,

$$f_\varphi(\mathbf{z}_t|\mathbf{z}_{t-1}) = \mathcal{N}(\psi(\mathbf{z}_{t-1}), \Sigma), \quad h_\phi(\mathbf{z}_t|\mathbf{x}_t) = \mathcal{N}(\gamma(\mathbf{x}_t), \Lambda). \tag{17}$$

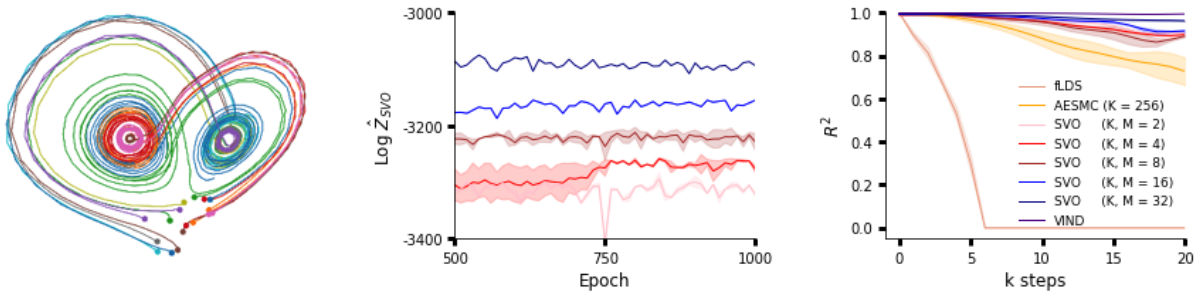

Figure 3. Summary of the Lorenz results: (left) latent trajectories inferred from nonlinear 10D observations; (center) $\log \hat{Z}_{SVO}$ as K, M increase (legend on the right). Larger K, M produce higher ELBO values; (right) R_k^2 on the dimensionality reduction task illustrating near-perfect reconstruction at 20 steps ahead on the validation set. Results averaged over 3 random seeds.

We define $\psi : \mathbb{R}^{d_{\mathbf{z}}} \to \mathbb{R}^{d_{\mathbf{z}}}$ and $\gamma : \mathbb{R}^{d_{\mathbf{x}}} \to \mathbb{R}^{d_{\mathbf{z}}}$ as nonlinear time invariant functions represented with deep neural networks. The covariances Σ and Λ are taken as time invariant trainable parameters or nonlinear functions of the latent space. This proposal choice allows the transition term of the inference network $f_\varphi(\mathbf{z}_t|\mathbf{z}_{t-1})$ to share the parameters φ defining $\{\psi, \Sigma\}$ with the transition term $f_\varphi(\mathbf{z}_t|\mathbf{z}_{t-1})$ of the target defined in Eq. (1) [19, 21, 25]. The evolution term of the variational posterior is exact, retaining both tractability and expressiveness.

The transition and emission densities are specified as follows:

$$f_\varphi(\mathbf{z}_t|\mathbf{z}_{t-1}) = \mathcal{N}(\psi(\mathbf{z}_{t-1}), \Sigma), \quad g_\theta(\mathbf{x}_t|\mathbf{z}_t) = \mathcal{N}(\upsilon(\mathbf{z}_t), \Gamma)\ . \tag{18}$$

The decoding term is defined using a deterministic nonlinear rate function $\upsilon : \mathbb{R}^{d_{\mathbf{z}}} \to \mathbb{R}^{d_{\mathbf{x}}}$ represented with a deep network and a noise model that need not be conjugate. Without loss of generality we consider a Gaussian emission density. The backward proposal defining the smoothing distribution below

$$q(\mathbf{z}_t|\mathbf{z}_{t+1}, \mathbf{x}_{1:T}) \propto r(\mathbf{z}_t|\zeta(\mathbf{z}_{t+1}))e(\mathbf{z}_t|\chi(\mathbf{x}_{1:T})), \tag{19}$$

is specified using nonlinear time invariant functions $\zeta : \mathbb{R}^{d_{\mathbf{z}}} \to \mathbb{R}^{d_{\mathbf{z}}}$ and $\chi : \mathbb{R}^{d_{\mathbf{x}}} \to \mathbb{R}^{d_{\mathbf{z}}}$ which we take as deep networks.

5 Experimental Results

In order to quantify the performance of the trained dynamics, we compute the k-step mean squared error (MSE) and its normalized version, the R_k^2. To do so, the trained transition function is applied to the latent state without any input data over a rolling window of k steps into the future. The emission function is then used to obtain a prediction $\hat{\mathbf{x}}_{t+k}$ which we compare with the observation $\mathbf{x}_{t+k}$.

$$\text{MSE}_k = \sum_{t=1}^{T-k} (\mathbf{x}_{t+k} - \hat{\mathbf{x}}_{t+k})^2 \tag{20}$$

$$R_k^2 = 1 - \frac{\text{MSE}_k}{\sum_{t=1}^{T-k} (\mathbf{x}_{t+k} - \bar{\mathbf{x}}_k)^2}\ ,$$

where $\bar{\mathbf{x}}_k$ is the average of $\mathbf{x}_{k+1:T}$. We note that the ELBO is not a performance statistic that generalizes across models. In contrast, the R_k^2 provides a metric to quantify the inferred dynamics. This procedure is defined in [11]. In all experiments, SVO is only given access to the observation sequence, and not the equations that govern the nonlinear systems. The latent trajectories and dynamics, transition, emission and encoding functions are all inferred.

Fitzhugh-Nagumo

The Fitzhugh-Nagumo (FN) system is a two dimensional simplification of the Hodgkin-Huxley model. The FN provides a geometric interpretation of the dynamics of spiking neurons and is described by two independent variables V_t and W_t with cubic and linear functions,

$$\begin{aligned} \dot{V} &= V - V^3/3 - W + I_{ext} \\ \dot{W} &= a(bV - cW). \end{aligned} \tag{21}$$

Eq. (21) was integrated over 200 time points with $I_{ext} = 1$ held constant and $a = 0.7, b = 0.8, c = 0.08$. The initial state was sampled uniformly over $[-3, 3]^2$ to generate 100 trials using 66 for training, 17 for validation and 17 for testing. We emphasize that dimensionality expansion is intrinsically harder than dimensionality reduction due to a loss of information. A one-dimensional Gaussian observation is defined on V_t with $\mathbf{x}_t = \mathcal{N}(V_t, 0.01)$. SVO is used to recover the two dimensional phase space and latent trajectories $\mathbf{z}_t = (V_t, W_t)$ of the original system. This task requires using information from future observations to correctly infer the initial state. Fig. 1 shows the results of the FN experiment. The left panel displays the original system. The center panel displays the learned dynamics and inferred trajectories on the test set using SVO to perform dimensionality expansion. Initial points (denoted with markers) located both inside and outside of the limit cycle in the original system are topologically invariant in the reconstruction. The right panel shows the R_k^2 comparison across models. AESMC with $K = 1024$ gives an $R_{30}^2 = 0.954$ in contrast to SVO with $K = 32, M = 32$ which gives an $R_{30}^2 = 0.993$. SVO outperforms AESMC and GfLDS.

Parameterizing the Transition Function

We study the effect of sharing the transition function between the proposal and target distribution. Fig. 2 illustrates the ELBO convergence as the number of particles K is increased. The left panel plots ELBO for SVO with network parameters shared between proposal and target. Increasing K produces a faster convergence and lower stochastic gradient noise. The center panel illustrates separate evolution networks for the proposal and the target. In contrast to sharing the transition function, separate evolution networks require a larger number of epochs for corresponding value of K. The ELBO obtains a lower value with larger stochastic gradient noise. The right panel juxtaposes shared and separate transition functions for $K = 16$ particles.

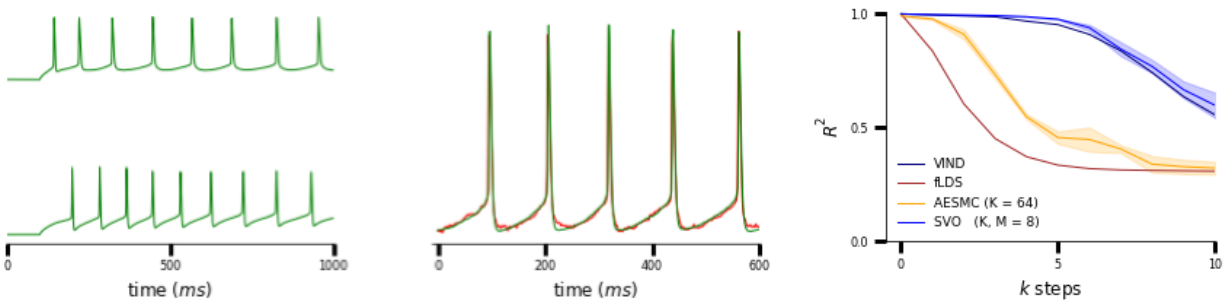

Figure 4. Summary of the Allen results: (left) two trials from the dataset illustrating different spiking dynamics; (center) the data against the predicted observation value using the dynamics learned over a rolling window ten steps ahead on the validation set. Hyperpolarization and depolarization nonlinearities are predicted by the inferred dynamics; (right) R_k^2 with $K, M = 8$ particles. SVO outperforms GfLDS and AESMC with $K = 64$. Results are averaged across 3 random seeds.

Lorenz Attractor

The Lorenz attractor is a chaotic nonlinear dynamical system defined by 3 independent variables,

$$\begin{aligned} \dot{z}_1 &= \sigma(z_2 - z_1)\,, \\ \dot{z}_2 &= z_1(\rho - z_3) - z_2\,, \\ \dot{z}_3 &= z_1 z_2 - \beta z_3\,. \end{aligned} \tag{22}$$

The system describes a mathematical model for atmospheric convection and produces nonlinear and non-periodic solutions. Eq. (22) is integrated over 250 time points with $\sigma = 10, \rho = 28, \beta = 8/3$ by generating randomized initial states in $[-10, 10]^3$. A $\mathbf{z}$-dependent neural network is used to produce ten dimensional nonlinear Gaussian observations with 100 trials, 66 for training, 17 for validation and 17 for testing. Fig. 3 provides the results of the Lorenz experiment. The left panel provides the inferred latent paths illustrating the attractor. The center plot provides $\log \hat{Z}_{SVO}$ as K, M increase (legend on the right). Larger K, M produce higher ELBO values. The right panel displays the R_k^2 comparison with $d_{\mathbf{z}} = 3$. Results are averaged over 3 random seeds. Increasing K, M produces R_k^2 improvements. SVO with $K, M = 2$ gives a higher R_k^2 than both GfLDS and AESMC using $K = 256$.

Single Cell Electrophysiology Data

Neuronal electrophysiology data was downloaded from the Allen Brain Atlas [13]. Intracellular voltage recordings from primary Visual Cortex of mouse, area layer 4 were collected. A step-function input current with an amplitude between 80 and 151pA was applied to each cell. A total of 40 trials from 5 different cells were split into 30 trials for training and 10 for validation. Each trial was divided into five parts and down-sampled from 10,000 time bins to 1,000 time bins in equal intervals. Each trial was normalized by its maximal value. Fig. 4 summarizes the Allen experiment. The left panel provides two trials of the 1D observations from the training set illustrating different spiking dynamics. The center panel plots the predicted observation using the dynamics learned over a rolling window ten steps ahead on the validation set. SVO captures hyperpolarization and depolarization nonlinearities when appying the inferred dynamics. The right panel displays the R_k^2 comparison with $d_{\mathbf{z}} = 3$. SVO outperforms AESMC and GfLDS.

6 Conclusion

We have introduced SVO, a framework for performing VI on state-space models jointly for hidden state inference and model parameter learning. SVO defines a novel backward simulation algorithm and approximate posterior obtained by sub-samping auxiliary random variables through a process of self normalized importance sampling. This augments the support of the proposal and boosts particle diversity. We have analyzed the resulting estimator theoretically and empirically, proving that SVO generates an unbiased estimate of the marginal likelihood constructed from the smoothed state sequence. Unlike standard particle smoothing methods, SVO simultaneously trains both the proposal and the target distribution from the observation sequence. SVO recovers nonlinear transition and emission functions in addition to latent states. Highlights include the ability to produce accurate long-range forecasts given smooth initial conditions from noisy, nonlinear differential equations using the trained latent dynamics. SVO consistently outperforms filtered objectives on all three experiments given fewer Monte Carlo samples. SVO is written in TensorFlow. An implementation is publicly available online.

Acknowledgements

We thank Daniel Hernandez Diaz and Christian Naesseth for helpful discussions and support in this project. We acknowledge funding from the NIH/NCI grant U54CA209997 and two NIH shared instrumentation grants, S10 OD012351 and S10 OD021764.

REFERENCES

[1] Christophe Andrieu, Arnaud Doucet, and Roman Holenstein, 'Particle markov chain monte carlo methods', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **72**(3), 269–342, (2010).

[2] Evan Archer, Il Memming Park, Lars Buesing, John Cunningham, and Liam Paninski. Black box variational inference for state space models, 2015.

[3] Mark Briers, Arnaud Doucet, and Simon Maskell, 'Smoothing algorithms for state–space models', *Annals of the Institute of Statistical Mathematics*, **62**(1), 61, (Jun 2009).

[4] Yuri Burda, Roger B. Grosse, and Ruslan Salakhutdinov, 'Importance weighted autoencoders', *CoRR*, **abs/1509.00519**, (2015).

[5] Tian Qi Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud, 'Neural ordinary differential equations', in *Advances in Neural Information Processing Systems 31*, eds., S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, 6571–6583, Curran Associates, Inc., (2018).

[6] Justin Domke and Daniel R Sheldon, 'Importance weighting and variational inference', in *Advances in Neural Information Processing Systems 31*, eds., S. Bengio, H. Wallach, H. Larochelle, K. Grauman,

N. Cesa-Bianchi, and R. Garnett, 4470–4479, Curran Associates, Inc., (2018).

[7] Yuanjun Gao, Evan Archer, Liam Paninski, and John P. Cunningham, 'Linear dynamical neural population models through nonlinear embedding', *NIPS 2016*, (2016).

[8] Simon J Godsill, Arnaud Doucet, and Mike West, 'Monte carlo smoothing for nonlinear time series', *Journal of the American Statistical Association*, **99**(465), 156–168, (2004).

[9] Pieralberto Guarniero, Adam Johansen, and Anthony Lee, 'The iterated auxiliary particle filter', *Journal of the American Statistical Association*, (08 2016).

[10] Jeremy Heng, Adrian N. Bishop, George Deligiannidis, and Arnaud Doucet. Controlled sequential monte carlo, 2017.

[11] Daniel Hernandez, A. Moretti, Ziqiang Wei, S. Saxena, John Cunningham, and Liam Paninski, 'A novel variational family for hidden nonlinear markov models', *CoRR*, **abs/1811.02459**, (2018).

[12] A. L. Hodgkin and A. F. Huxley, 'A quantitative description of membrane current and its application to conduction and excitation in nerve', *Bulletin of Mathematical Biology*, **52**(1), 25–71, (Jan 1990).

[13] Allan R. Jones, Caroline C. Overly, and Susan M. Sunkin, 'The allen brain atlas: 5 years and beyond', *Nature Reviews Neuroscience*, **10**, 821 EP, (10 2009).

[14] Diederik P. Kingma and Max Welling, 'Auto-encoding variational bayes', *CoRR*, **abs/1312.6114**, (2013).

[15] Genshiro Kitagawa, 'Monte carlo filter and smoother for non-gaussian nonlinear state space models', *Journal of computational and graphical statistics*, **5**(1), 1–25, (1996).

[16] Rahul G. Krishnan, Uri Shalit, and David Sontag. Deep kalman filters, 2015.

[17] Dietrich Lawson, George Tucker, Dai Bo, and Ragesh Raganath, 'Revisiting auxiliary latent variables in generative models', *ICLR Workshops*, (2019).

[18] Dietrich Lawson, George Tucker, Christian Naesseth, Chris Maddison, Ryan Adams, and Yee Teh, 'Twisted variational sequential monte carlo', *Bayesian Deep Learning Workshop, NIPS*, (2016).

[19] Tuan Anh Le, Maximilian Igl, Tom Rainforth, Tom Jin, and Frank Wood, 'Auto-encoding sequential monte carlo', in *International Conference on Learning Representations*, (2018).

[20] Fredrik Lindsten, Jouni Helske, and Matti Vihola, 'Graphical model inference: Sequential monte carlo meets deterministic approximations', in *Advances in Neural Information Processing Systems*, pp. 8190–8200, (2018).

[21] Chris J Maddison, John Lawson, George Tucker, Nicolas Heess, Mohammad Norouzi, Andriy Mnih, Arnaud Doucet, and Yee Teh, 'Filtering variational objectives', in *Advances in Neural Information Processing Systems 30*, eds., I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, 6573–6583, Curran Associates, Inc., (2017).

[22] Antonio Moretti, Andrew Stirn, Gabriel Marks, and Itsik Pe'er, 'Autoencoding topographic factors', *Journal of Computational Biology*, **26**(6), 546–560, (2019). PMID: 30526005.

[23] Antonio Moretti, Zizhao Wang, Luhuan Wu, and Itsik Pe'er, 'Smoothing nonlinear variational objectives with sequential monte carlo', *ICLR Workshops*, (2019).

[24] Antonio Khalil Moretti, Zizhao Wang, Luhuan Wu, Iddo Drori, and Itsik Pe'er, 'Particle smoothing variational objectives', *CoRR*, **abs/1909.09734**, (2019).

[25] Christian Naesseth, Scott Linderman, Rajesh Ranganath, and David Blei, 'Variational sequential monte carlo', in *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, eds., Amos Storkey and Fernando Perez-Cruz, volume 84 of *Proceedings of Machine Learning Research*, pp. 968–977, Playa Blanca, Lanzarote, Canary Islands, (09–11 Apr 2018). PMLR.

[26] Christian A. Naesseth, Fredrik Lindsten, and Thomas B. Schn. Elements of sequential monte carlo, 2019.

[27] Chethan Pandarinath, Daniel J. O'Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D. Stavisky, Jonathan C. Kao, Eric M. Trautmann, Matthew T. Kaufman, Stephen I. Ryu, Leigh R. Hochberg, Jaimie M. Henderson, Krishna V. Shenoy, Larry F. Abbott, and David Sussillo. Inferring single-trial neural population dynamics using sequential auto-encoders, 2017.

[28] Adam Persing and Ajay Jasra, 'Likelihood computation for hidden Markov models via generalized two-filter smoothing', *Statistics & Probability Letters*, **83**(5), 1433–1442, (2013).

[29] Yuan Zhao, Josue Nassar, Ian Jordan, Mnica Bugallo, and Il Memming Park. Streaming variational monte carlo, 2019.