Free-Form Variational Inference for Gaussian Process State-Space Models

Xuhui Fan [1] **Edwin V. Bonilla** [2] **Terence J. O'Kane** [3] **Scott A. Sisson** [4]

Abstract

Gaussian process state-space models (GPSSMs) provide a principled and flexible approach to modeling the dynamics of a latent state, which is observed at discrete-time points via a likelihood model. However, inference in GPSSMs is computationally and statistically challenging due to the large number of latent variables in the model and the strong temporal dependencies between them. In this paper, we propose a new method for inference in Bayesian GPSSMs, which overcomes the drawbacks of previous approaches, namely over-simplified assumptions, and high computational requirements. Our method is based on free-form variational inference via stochastic gradient Hamiltonian Monte Carlo within the inducing-variable formalism. Furthermore, by exploiting our proposed variational distribution, we provide a collapsed extension of our method where the inducing variables are marginalized analytically. We also showcase results when combining our framework with particle MCMC methods. We show that, on six real-world datasets, our approach can learn transition dynamics and latent states more accurately than competing methods.

1. Introduction

State-space models (SSMs; Murphy, 2023, Ch. 29) characterize the underlying dynamics of a latent state given a set of observations via a transition (or evolution) function and an observation model. As a modeling framework, they provide a general approach for understanding time-series data (Kitagawa, 1987) and for data assimilation problems (Katzfuss et al., 2016). Applications of SSMs abound and span diverse areas such as econometrics (Tsay, 2005), meteorology (Hernandez et al., 1991), control engineering (Ogata et al., 2010) and neuroscience (Brown et al., 1998).

In this paper, we focus on Bayesian SSMs, where the transition function describing the dynamics of the system is given a prior distribution. A paradigmatic example of Bayesian SSMs are Gaussian process state-space models (GPSSMs; Frigola, 2015), where this prior distribution is a Gaussian process (GP; Williams & Rasmussen, 2006). Due to their Bayesian non-parametric nature, GPSSMs represent a principled and flexible approach to Bayesian SSMs.

However, the flexibility of Gaussian processes (GPs) adds significant computational and statistical challenges to the already difficult problem of inference in Bayesian SSMs. Indeed, even for non-Bayesian SSMs, standard problems such as filtering, smoothing, and prediction are, in general, analytically intractable[1]. Having a GP prior over the transition function in SSMs increases the number of latent variables significantly; incorporates strong (and potentially long-term) dependencies across states; and introduces a cubic time complexity as a function of the number of observations.

Within the GP community, significant advances have been made addressing the computational issues in GP regression and classification problems, most notably using inducing-variable approximations (Titsias, 2009; Hensman et al., 2015; Rossi et al., 2021) but also random-feature expansions (Cutajar et al., 2017; Marmin & Filippone, 2022) and, more recent innovative approaches such as the Vecchia approximation (Sauer et al., 2022). Although these approximations are applicable to GP-based dynamic models, the challenges above remain prevalent within the context of GPSSMs.

Nevertheless, previous approaches have developed insightful and practical algorithms for inference in GPSSMs, mainly based on inducing-variable approximations, which is also our main underpinning methodology for scalable GPs. In outlining the most relevant approaches, our main object of interest is the approximate joint posterior over state trajectories $\mathbf{x}_{0:T}$ and inducing variables $\mathbf{u}$, $q(\mathbf{x}_{0:T}, \mathbf{u})$, where $T+1$ is the length of the trajectory. We consider two main aspects of this joint distribution: (i) whether the dependencies between state trajectories and inducing variables are

[1]University of Newcastle, Australia [2]CSIRO's Data61, Australia [3]CSIRO's Environment, Australia [4]University of New South Wales, Australia. Correspondence to: Xuhui Fan <xhfan.ml@gmail.com>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

[1]With the notable exception of linear state transition and observation models with additive Gaussian noise, for which the solution to the optimal filtering problem is given by the Kalman filter (Kalman, 1960).

captured and (ii) whether their corresponding distributions are unconstrained, i.e., not restricted to a sub-optimal parametric form. The seminal variational Gaussian process state space model (VGPSSM) proposed by Frigola et al. (2014) as well as the subsequent identifiable Gaussian process state space model (IGPSSM) of Eleftheriadis et al. (2017) use mean-field approaches, therefore, ignoring the posterior dependencies between state trajectories and inducing variables. The more recent methods, namely the probabilistic recurrent state space model (PRSSM) of Doerr et al. (2018) and the variationally coupled dynamic trajectories (VCDT) of Ialongo et al. (2019) introduce couplings across state trajectories and inducing variables. However, their posteriors are constrained to be Gaussians. Thus, these previous works have either assumed independence between state trajectories and inducing variables or imposed strong parametric constraints in their corresponding posteriors or both.

As shown by Ialongo et al. (2019), a mean-field posterior can yield poor practical performance. Similarly, a Gaussian assumption on the state posterior or the inducing variable posterior is also very strong and, by definition, will not generally capture the true posterior even in the limit of infinite computation. To address these issues, we propose a free-form variational inference approach to posterior estimation in GPSSMs that models the full joint distribution over states and inducing variables, $q(\mathbf{x}_{0:T}, \mathbf{u})$, without any mean-field or parametric assumptions. We refer to our method as free-form variational dynamics (FFVD) and summarize the major differences between its posterior assumptions and those of previous approaches in Table 1. Below we describe our contributions in more detail.

(i) Flexible posterior: We develop an inference algorithm for GPSSMs based on stochastic gradient Hamiltonian Monte Carlo (SGHMC; Chen et al., 2014; Havasi et al., 2018), which represents the posterior over states and inducing variables using samples. Further, our approach lifts the limitations of previous variational approaches to GPSSMs, as they have ignored couplings in the posterior between the inducing variables and the state trajectories or have assumed a constrained parametric form for this posterior. Regarding this, our approach naturally capture the posterior correlations between states and inducing variables; does not constrain this posterior to any parametric form, and, is scalable to a large number of observations.

(ii) Collapsed inference that accelerates convergence: we show that (i) our formulation allows us to collapse the inducing variables $\mathbf{u}$, (ii) sample from the lower-dimensional marginal $q(\mathbf{x}_{0:T})$ and, at the end of the sampling procedure, (iii) obtain samples from the conditional $q(\mathbf{u} \mid \mathbf{x}_{0:T})$, for which we derive a closed-form expression. We show that collapsing accelerates convergence significantly.

(iii) Extensions with particle Markov chain Monte Carlo (PMCMC): we further investigate whether more elaborate inference algorithms, such as PMCMC (Andrieu et al., 2010), that account for the sequential nature of the problem can provide more accurate posteriors.

(iv) State-of-the-art performance: we showcase the properties and benefits of our approach compared to previous methods such as VGPSSM, PRSSM and VCDT in a synthetic example and six system identification benchmarks. Overall, our method provides state-of-the-art performance when evaluated on these problems, while having comparable computational requirements to previous approaches.

We open-source our proposed FFVD approach, together with baseline methods reported in the experiments: `https://github.com/xuhuifan/FFVD`.

2. Inference in Gaussian Process Models

Gaussian processes (GPs) are priors over functions where every subset of function values follow a Gaussian distribution. We use $f(\mathbf{x}) \sim \mathcal{GP}\left(m_f(\mathbf{x}), \kappa_f(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta})\right)$ to denote that f is distributed according to a GP with mean function $m_f(\cdot)$ and covariance function $\kappa_f(\cdot, \cdot; \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ are referred to as the GP hyper-parameters[2]. By definition, a GP prior over functions implies a finite prior over T function values $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_T)]^\top$, i.e., $\mathbf{f} \sim \mathcal{N}(\mathbf{f}; \mathbf{m}, \mathbf{K})$, where $\mathbf{m}$ and $\mathbf{K}$ are obtained by evaluating the mean function and covariance function at all the inputs $\mathbf{X} := \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$.

In supervised learning settings, we are given input-output observations $\{\mathbf{x}_t, y_t\}_{t=1}^T$ and a conditional likelihood model $p(\mathbf{y} \mid \mathbf{f})$. Inference involves estimating the posterior distribution $p(\mathbf{f} \mid \mathbf{y}, \mathbf{X})$ and the hyper-parameters $\boldsymbol{\theta}$ from data. Consequently, we can use these to estimate the posterior predictive distribution at a new point $\mathbf{x}_\star$, i.e., $p(f(\mathbf{x}_\star) \mid \mathbf{X}, \mathbf{y}, \mathbf{x}_\star)$. Notoriously, these tasks have cubic time complexity as a function of the number of training observations, i.e., $\mathcal{O}(T^3)$, arising from algebraic operations involving the computation of the inverse covariance and its log determinant. This motivates the need for sparse approximations.

2.1. Sparse GP Approximations via Inducing Variables

Various approximations have been proposed to deal with the cubic computational complexity of inference in GP models. In this paper, we focus on inducing-variable approximations based on variational inference, as originally proposed by Titsias (2009) and made scalable to very large datasets by Hensman et al. (2013). The main idea of these approximations is to augment the space of function values with a set of M inducing variables $\mathbf{u} := \{u_i\}$ and their correspond-

[2]We note here that the mean function can also have additional parameters but, in this work, we have simply assumed identity mean functions.

Table 1. Comparison across methods in terms of their assumptions on the variational distribution. The rows refer to whether the variational posterior captures the dependencies between the state trajectories and the inducing variables (coupled $q(\mathbf{x}_{0:T}, \mathbf{u})$); the distribution over states is unconstrained (unconstrained $q(\mathbf{x}_{0:T} \mid \mathbf{u})$ or $q(\mathbf{x}_{0:T})$), i.e., not restricted to a sub-optimal parametric form; and whether the distribution over the inducing variables is also unconstrained (unconstrained $q(\mathbf{u})$). Our method is referred to as FFVD.

	VGPSSM	IGPSSM	PRSSM	VCDT	FFVD
Coupled $q(\mathbf{x}_{0:T}, \mathbf{u})$	✗	✗	✓	✓	✓
Unconstrained $q(\mathbf{x}_{0:T} \mid \mathbf{u})$ or $q(\mathbf{x}_{0:T})$	✓	✗	✗	✗	✓
Unconstrained $q(\mathbf{u})$	✓	✗	✗	✗	✓

ing inducing inputs $\mathbf{Z} := \{\mathbf{z}_i\}$. Thus, inference involves estimating the posterior $q(\mathbf{u}, \mathbf{f}) \approx p(\mathbf{u}, \mathbf{f} \mid \mathbf{X}, \mathbf{y})$ and the GP hyper-parameters $\boldsymbol{\theta}$ via variational inference. Under the assumption that $q(\mathbf{u}, \mathbf{f}) := q(\mathbf{u})p(\mathbf{f} \mid \mathbf{u})$ where $p(\mathbf{f} \mid \mathbf{u})$ is the conditional prior, the variational objective, the so-called evidence lower bound (ELBO), decomposes over the observations, and inference can be carried out with time complexity of $\mathcal{O}(M^3)$, providing significant advantages when $M \ll T$.

3. Gaussian Process State-Space Models

Let us assume we are given a time series of T multi-dimensional observations $\mathbf{y}_{1:T}$ and denote their corresponding latent states with $\mathbf{x}_{0:T}$, where $\mathbf{x}_0$ is the initial state. Here we denote the time series $\mathbf{x}_{t_1:t_2} := \{\mathbf{x}_{t_1}, \ldots, \mathbf{x}_{t_2}\}$, and similarly for $\mathbf{y}_{1:T}$. In general, $\mathbf{x}_t \in \mathbb{R}^{d_x}$ and $\mathbf{y}_t \in \mathbb{R}^{d_y}$. Gaussian process state-space models (GPSSMs) formulate a discrete-time state-space model (SSM) where the transition dynamics is given by a GP. The full generative process is:

$$\mathbf{x}_0 \sim p(\mathbf{x}_0),\ f(\mathbf{x}) \sim \mathcal{GP}(m_f(\mathbf{x}), \kappa_f(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta})), \quad (1)$$

$$\mathbf{f}_t := f(\mathbf{x}_{t-1}), \quad \mathbf{x}_t \mid \mathbf{f}_t \sim \mathcal{N}(\mathbf{x}_t; \mathbf{f}_t, \mathbf{Q}), \quad (2)$$

$$\mathbf{y}_t \mid \mathbf{x}_t \sim p(\mathbf{y}_t \mid \mathbf{x}_t, \boldsymbol{\phi}), \quad (3)$$

where $\mathbf{Q}$ is the transition process covariance, and $\boldsymbol{\phi}$ is the vector of parameters of the conditional likelihood $p(\mathbf{y}_t \mid \mathbf{x}_t, \boldsymbol{\phi})$. Although our framework does not make any parametric assumptions about this conditional likelihood, in our experiments in § 7, we adopt the same setting as in previous works (Ialongo et al., 2019; Doerr et al., 2018) and set $p(\mathbf{y}_t \mid \mathbf{x}_t, \boldsymbol{\phi}) := \mathcal{N}(\mathbf{y}_t; \mathbf{C}\mathbf{x}_t + \mathbf{d}, \mathbf{R})$, with $\boldsymbol{\phi} = \{\mathbf{C}, \mathbf{d}, \mathbf{R}\}$, where $\mathbf{C}, \mathbf{d}$ are the weights and bias of the linear transformation on $\mathbf{x}_t$ and $\mathbf{R}$ is the observation covariance.

3.1. Joint Distribution

As shown by Frigola (2015), to sample $\mathbf{f}_t$, instead of conditioning on an infinite-dimensional function, we can condition only on the transitions seen up to (but not including) time t, i.e., $\{(\mathbf{x}_{i-1}, \mathbf{f}_i)\}_{i=1}^{t-1}$. We can then write the joint distribution over latent variables and observations as:

$$p(\mathbf{y}_{1:T}, \mathbf{x}_{0:T}, \mathbf{f}_{1:T}) = p(\mathbf{x}_0) \prod_{t=1}^{T} p(\mathbf{f}_t \mid \mathbf{f}_{1:t-1}, \mathbf{x}_{0:t-1}) p(\mathbf{x}_t \mid \mathbf{f}_t) p(\mathbf{y}_t \mid \mathbf{x}_t), \quad (4)$$

where $p(\mathbf{x}_0), p(\mathbf{x}_t \mid \mathbf{f}_t), p(\mathbf{y}_t \mid \mathbf{x}_t)$ are defined as above and, for the edge case of $t = 1$, we have: $p(\mathbf{f}_1 \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{f}_1; m_f(\mathbf{x}_0), \kappa_f(\mathbf{x}_0, \mathbf{x}_0; \boldsymbol{\theta}))$. Furthermore, we recognize each conditional distribution $p(\mathbf{f}_t \mid \mathbf{f}_{1:t-1}, \mathbf{x}_{0:t-1})$ in Eq. (4) as the GP prediction at a single point $\mathbf{x}_{t-1}$ using noiseless outputs $\mathbf{f}_{1:t-1}$ and inputs $\mathbf{x}_{0:t-2}$,

$$p(\mathbf{f}_t \mid \mathbf{f}_{1:t-1}, \mathbf{x}_{0:t-1}) = p(\mathbf{f}_t \mid \mathbf{x}_{t-1}, \mathbf{f}_{1:t-1}, \mathbf{x}_{0:t-2}) \quad (5)$$

$$= \mathcal{N}(\mathbf{f}_t; \boldsymbol{\mu}_f, \boldsymbol{\Sigma}_f), \quad (6)$$

with conditional mean and covariance given by

$$\boldsymbol{\mu}_f = \mathbf{m}_{t-1} + \mathbf{K}_{t-1,0:t-2}\mathbf{K}_{0:t-2}^{-1}(\mathbf{f}_{1:t-1} - \mathbf{m}_{0:T-2}) \quad (7)$$

$$\boldsymbol{\Sigma}_f = \mathbf{K}_{t-1} - \mathbf{K}_{t-1,0:t-2}\mathbf{K}_{0:t-2}^{-1}\mathbf{K}_{0:t-2,t-1}), \quad (8)$$

where the subscript notation indicates the mean vectors and covariance matrices obtained from evaluating the mean function and covariance function, respectively, at the corresponding ranges, $\mathbf{m}_{t_1:t_2} := m_f(\mathbf{x}_{t_1:t_2})$, $\mathbf{m}_t := \mathbf{m}_{t:t}$, $\mathbf{K}_{t_1:t_2,t_3:t_4} := \kappa_f(\mathbf{x}_{t_1:t_2}, \mathbf{x}_{t_3:t_4}; \boldsymbol{\theta})$, $\mathbf{K}_{t_1:t_2} := \mathbf{K}_{t_1:t_2,t_1:t_2}$ and $\mathbf{K}_t := \mathbf{K}_{t:t}$.

3.2. Multidimensional Latent States & Control Inputs

In the case of multidimensional latent states, i.e., $d_x > 1$, we assume independent GPs on each dimension, each with its own mean function and covariance function. Because of this independence assumption, each GP only has to condition on its own function evaluations. Therefore, for simplicity in the notation, we do not index the means and covariances with respect to their dimension d and consider the underlying GPs, their means, and covariance functions as multi-dimensional. Furthermore, as we shall see in § 7, our experiments consider additional control inputs ($\mathbf{a}_t$) that affect the transitions in a Markovian way. This is easy to incorporate in our framework by augmenting our input space and, therefore, indexing the GPs in the higher-dimensional space given by

the concatenation $[\mathbf{x}^\top, \mathbf{a}^\top]^\top$. However, since the control inputs are fixed and deterministic, we do not need to include them as part of out inference method and do not make them explicit in our subsequent mathematical development.

3.3. Sparse GPSSM

In order to make the GPSSM inherently scalable, similarly to the standard supervised regression setting with GPs described in § 2, we augment the full GPSSM model with M inducing variables and corresponding inducing inputs $\mathbf{u} := \{u_i\}_{i=1}^M$ and $\mathbf{Z} := \{\mathbf{z}_i\}_{i=1}^M$. This gives rise to the sparse GPSSM model:

$$p(\mathbf{y}_{1:T}, \mathbf{x}_{0:T}, \mathbf{f}_{1:T}, \mathbf{u} \,|\, \mathbf{Z}) = p(\mathbf{u} \,|\, \mathbf{Z}) p(\mathbf{x}_0) \prod_{t=1}^{T} p(\mathbf{f}_t \,|\, \mathbf{f}_{1:t-1}, \mathbf{x}_{0:t-1}, \mathbf{u}, \mathbf{Z}) p(\mathbf{x}_t \,|\, \mathbf{f}_t) p(\mathbf{y}_t \,|\, \mathbf{x}_t), \quad (9)$$

where the prior over the inducing variables is determined by the GP prior, i.e., $p(\mathbf{u} \,|\, \mathbf{Z}) = \mathcal{N}(\mathbf{m}_Z, \mathbf{K}_Z)$ with $\mathbf{m}_Z := m_f(\mathbf{Z})$ and $\mathbf{K}_Z := \kappa_f(\mathbf{Z}, \mathbf{Z}; \boldsymbol{\theta})$. Besides having a prior over the inducing variables $\mathbf{u}$, the main difference with our previous full-model formulation of § 3.1 is that the conditional distributions over $\mathbf{f}_t$ have been augmented with inducing variables $\mathbf{u}$ and corresponding inducing inputs $\mathbf{Z}$. Each conditional, $p(\mathbf{f}_t \,|\, \mathbf{f}_{1:t-1}, \mathbf{x}_{0:t-1}, \mathbf{u}, \mathbf{Z}) = p(\mathbf{f}_t \,|\, \mathbf{x}_{t-1}, \mathbf{f}_{1:t-1}, \mathbf{u}, \mathbf{x}_{0:t-2}, \mathbf{Z})$, is the predictive (GP regression) distribution of $\mathbf{f}_t$ at test input $\mathbf{x}_{t-1}$ when observing $(\mathbf{f}_{1:t-1}, \mathbf{u})$ at their respective locations $(\mathbf{x}_{0:t-2}, \mathbf{Z})$.

4. Free-Form Variational Inference

In this section, we develop a posterior estimation method using variational inference (VI). We show that under a specific form of the approximate posterior, we can estimate the joint posterior over $\{\mathbf{x}_{0:T}, \mathbf{u}\}$ in free-form, i.e., optimally, without making any assumptions such as independence between state trajectories and inducing variables, typical of mean-field approaches (Frigola et al., 2014; Eleftheriadis et al., 2017), or constraining the form of the posterior to sub-optimal parametric forms (Ialongo et al., 2019; Doerr et al., 2018).

4.1. Variational Family and Evidence Lower Bound

Variational inference is underpinned by the maximization of the evidence lower bound (ELBO), which is equivalent to minimizing the Kullback-Leibler (KL) divergence between the approximate posterior and the true posterior. This objective is given by

$$\mathcal{L}_{\text{ELBO}}(q) := \mathbb{E}_{q(\mathbf{x}_{0:T}, \mathbf{u}, \mathbf{f}_{1:T})} \left[\log p(\mathbf{y}_{1:T}, \mathbf{x}_{0:T}, \mathbf{f}_{1:T}, \mathbf{u} \,|\, \mathbf{Z}) - \log q(\mathbf{x}_{0:T}, \mathbf{u}, \mathbf{f}_{1:T}) \right], \quad (10)$$

where $p(\mathbf{y}_{1:T}, \mathbf{x}_{0:T}, \mathbf{f}_{1:T}, \mathbf{u} \,|\, \mathbf{Z})$ is the joint distribution in Eq. (9) and $q(\mathbf{x}_{0:T}, \mathbf{u}, \mathbf{f}_{1:T})$ is our proposed approximate joint posterior, which we define as

$$q(\mathbf{f}_{1:T} \,|\, \mathbf{x}_{0:T}, \mathbf{u}) := \prod_{t=1}^{T} p(\mathbf{f}_t \,|\, \mathbf{f}_{1:t-1}, \mathbf{x}_{0:t-1}, \mathbf{u}, \mathbf{Z}), \quad (11)$$

$$q(\mathbf{x}_{0:T}, \mathbf{u}, \mathbf{f}_{1:T}) := q(\mathbf{x}_{0:T}, \mathbf{u}) q(\mathbf{f}_{1:T} \,|\, \mathbf{x}_{0:T}, \mathbf{u}). \quad (12)$$

It is easy to show that the ELBO, as defined in Eq. (10), is a lower bound on the log marginal likelihood, i.e., $\mathcal{L}_{\text{ELBO}}(q) \leq \log p(\mathbf{y} \,|\, \mathbf{Z})$. We use the prior distribution of $\mathbf{f}_{1:T}$ as its variational distribution in Eq. (11), which follows the same settings as all previous scalable variational approaches (that we are aware of) VGPSSM, IGPSSM, PRSSM and VCDT. It is necessary to make the variational framework scalable as a function of the number of observations, which will yield an inference algorithm that alleviates the cubic time complexity on the number of observations (T) and, therefore, scales up to large datasets. Crucially, our proposed joint variational distribution in Eq. (12), will allow us to derive an *optimal* variational distribution $q(\mathbf{x}_{0:T}, \mathbf{u})$ in free-form, without imposing any parametric constraints over it and, instead, represent it via samples. We will describe this in the next section.

4.2. Evidence Lower Bound Maximization

Our first step is to expand the expression for the ELBO in Eq. (10) using our joint model distribution in Eq. (9) and our proposed variational distribution in Eqs. (11) and (12). We first note that our definition of the variational distribution in Eq. (12) uses the same conditional prior as in the joint distribution in Eq. (9). Therefore, this term cancels out, avoiding the computation of operations on fully-coupled high-dimensional distributions over latent functions. Thus, we have that:

$$\mathcal{L}_{\text{ELBO}}(q) = \int q(\mathbf{x}_{0:T}, \mathbf{u}) \Bigg\{ - \log q(\mathbf{x}_{0:T}, \mathbf{u}) + \log \left[p(\mathbf{x}_0) p(\mathbf{u} \,|\, \mathbf{Z}) \prod_{t=1}^{T} p(\mathbf{y}_t \,|\, \mathbf{x}_t) \right] + \mathbb{E}_{q(\mathbf{f}_{1:T} \,|\, \mathbf{x}_{0:T}, \mathbf{u})} \left[\log \prod_{t=1}^{T} p(\mathbf{x}_t \,|\, \mathbf{f}_t) \right] \Bigg\} \mathrm{d}\mathbf{x}_{0:T} \mathrm{d}\mathbf{u}. \quad (13)$$

4.2.1. ELBO Maximization

Next we aim to maximize the ELBO functional above with respect to $q(\mathbf{x}_{0:T-1}, \mathbf{u})$ subject to the constraint $\int q(\mathbf{x}_{0:T-1}, \mathbf{u}) \, \mathrm{d}\mathbf{x}_{0:T-1} \mathrm{d}\mathbf{u} = 1$. We can do this by solving

the corresponding Euler-Lagrange equation:

$$\frac{\partial}{\partial q(\mathbf{x}_{0:T}, \mathbf{u})} \Bigg\{ - q(\mathbf{x}_{0:T}, \mathbf{u}) \log q(\mathbf{x}_{0:T}, \mathbf{u}) + q(\mathbf{x}_{0:T}, \mathbf{u}) \log \Big[p(\mathbf{x}_0) p(\mathbf{u} \,|\, \mathbf{Z}) \prod_{t=1}^{T} p(\mathbf{y}_t \,|\, \mathbf{x}_t) \Big] + q(\mathbf{x}_{0:T}, \mathbf{u}) \mathbb{E}_{q(\mathbf{f}_{1:T} \,|\, \mathbf{x}_{0:T}, \mathbf{u})} \Big[\log \prod_{t=1}^{T} p(\mathbf{x}_t \,|\, \mathbf{f}_t) \Big] \Bigg\} = 0. \tag{14}$$

By doing the corresponding derivatives we obtain

$$- \log q(\mathbf{x}_{0:T}, \mathbf{u}) - 1 + \log \Big[p(\mathbf{x}_0) p(\mathbf{u} \,|\, \mathbf{Z}) \prod_{t=1}^{T} p(\mathbf{y}_t \,|\, \mathbf{x}_t) \Big] + \mathbb{E}_{q(\mathbf{f}_{1:T} \,|\, \mathbf{x}_{0:T}, \mathbf{u})} \Big[\log \prod_{t=1}^{T} p(\mathbf{x}_t \,|\, \mathbf{f}_t) \Big] = 0. \tag{15}$$

Now we note that the expectation above can be solved in closed form:

$$\begin{aligned} &\mathbb{E}_{q(\mathbf{f}_{1:T} \,|\, \mathbf{x}_{0:T}, \mathbf{u})} \Big[\log \prod_{t=1}^{T} p(\mathbf{x}_t \,|\, \mathbf{f}_t) \Big] \\ &= \sum_{t=1}^{T} \mathbb{E}_{p(\mathbf{f}_t \,|\, \mathbf{x}_{t-1}, \mathbf{u}, \mathbf{Z})} \log p(\mathbf{x}_t \,|\, \mathbf{f}_t) \\ &= \sum_{t=1}^{T} \Big[\log \mathcal{N}(\mathbf{x}_t; \mathbf{u}_{x_t}, \mathbf{Q}) - \frac{1}{2} \operatorname{Tr}(\mathbf{Q}^{-1} \mathbf{B}_{t-1}) \Big], \end{aligned} \tag{16}$$

where $p(\mathbf{f}_t \,|\, \mathbf{x}_{t-1}, \mathbf{u}, \mathbf{Z})$ is the GP predictive distribution over the function values $\mathbf{f}_t$ at locations $\mathbf{x}_{t-1}$ given the inducing variables $\mathbf{u}$ at inducing inputs $\mathbf{Z}$, i.e.,

$$p(\mathbf{f}_t \,|\, \mathbf{x}_{t-1}, \mathbf{u}, \mathbf{Z}) = \mathcal{N}(\mathbf{f}_t; \boldsymbol{\mu}_{x_t}, \mathbf{B}_{t-1}), \tag{17}$$

$$\boldsymbol{\mu}_{x_t} := \mathbf{m}_{t-1} + \mathbf{A}_{t-1}(\mathbf{u} - \mathbf{m}_Z). \tag{18}$$

Here we have defined

$$\mathbf{A}_{t-1} := \mathbf{K}_{t-1,Z} \mathbf{K}_Z^{-1}, \tag{19}$$

$$\mathbf{B}_{t-1} := \mathbf{K}_{t-1} - \mathbf{K}_{t-1,Z} \mathbf{K}_Z^{-1} \mathbf{K}_{Z,t-1}, \tag{20}$$

and the cross-covariance term $\mathbf{K}_{t-1,Z} := \kappa_f(\mathbf{x}_{t-1}, \mathbf{Z}; \boldsymbol{\theta})$ and similarly for $\mathbf{K}_{Z,t-1}$. Finally, $\operatorname{Tr}(\cdot)$ is the trace operator and, as defined at the beginning of § 3, $\mathbf{Q}$ is the transition noise covariance.

4.2.2. Optimal Variational Posterior

With this, we obtain the form of the optimal variational distribution $q^*(\mathbf{u}, \mathbf{x}_{0:T})$ up to a normalizing constant Z_q as:

$$\log q^*(\mathbf{u}, \mathbf{x}_{0:T}) = \log p(\mathbf{u} \,|\, \mathbf{Z}) + \log p(\mathbf{x}_0) + \sum_{t=1}^{T} \Big[\log p(\mathbf{y}_t \,|\, \mathbf{x}_t) + \log \mathcal{N}(\mathbf{x}_t; \boldsymbol{\mu}_{x_t}, \mathbf{Q}) - \frac{1}{2} \operatorname{Tr}(\mathbf{Q}^{-1} \mathbf{B}_{t-1}) \Big] + \log Z_q. \tag{21}$$

Here we note that the function values $\mathbf{f}_{1:T}$ have, effectively, been marginalized variationally. The optimal joint posterior over inducing variables and state trajectories depends on the prior over the inducing variables $p(\mathbf{u} \,|\, \mathbf{Z})$ stemming from the GP functional prior, the prior over the initial state $p(\mathbf{x}_0)$ and the conditional likelihood terms $p(\mathbf{y}_t \,|\, \mathbf{x}_t)$. It also depends on the resulting transitions mapping $\mathbf{x}_{t-1}$ to $\mathbf{x}_t$ via the densities $\mathcal{N}(\mathbf{x}_t; \boldsymbol{\mu}_{x_t}, \mathbf{Q})$, where $\boldsymbol{\mu}_{x_t}$ depends on $\mathbf{x}_{t-1}$ in a nonlinear way, as specified by Eq. (19). The final trace term, $\operatorname{Tr}(\mathbf{Q}^{-1} \mathbf{B}_{t-1})$, can be seen as a regularization term acting on state transitions, encouraging higher transition variances and, therefore, helping prevent overfitting.

4.2.3. Alternative Perspective

An alternative way to obtain the optimal joint posterior over state trajectories and inducing variable is by bounding the true log joint marginal $\log p(\mathbf{y}_{1:T}, \mathbf{x}_{0:T}, \mathbf{u} \,|\, \mathbf{Z})$ using Jensen's inequality. We give details of such an approach in Appendix E. This has been used by previous work in standard regression settings (see, e.g., Rossi et al., 2021, and references therein). However, our setting considers the more complex case of GPSSMs. Additionally, our development in this section shows much more clearly the optimal nature of the variational posterior, as we have obtained it via calculus of variations.

4.3. Posterior Sampling

Having the form of the optimal posterior in Eq. (21), we can then set the latent variables $\boldsymbol{\Psi} := \{\mathbf{u}, \mathbf{x}_{0:T}\}$ and have $\tilde{q}(\boldsymbol{\Psi}) \propto q(\boldsymbol{\Psi}) \approx p(\boldsymbol{\Psi} \,|\, \mathbf{y}_{1:T})$. Thus, we can draw samples from our approximate posterior using stochastic gradient Hamiltonian Monte Carlo (SGHMC; Chen et al., 2014; Havasi et al., 2018) and the energy function $U(\boldsymbol{\Psi}) = -\log p(\boldsymbol{\Psi}, \mathbf{y}_{1:T}) = -\log p(\boldsymbol{\Psi} \,|\, \mathbf{y}_{1:T}) + \log Z_q \approx -\log \tilde{q}(\boldsymbol{\Psi}) + \log Z_q$. Using this procedure, samples from the target distribution can be obtained even with noisy gradients (e.g., with mini-batches) without requiring the evaluation of Metropolis ratios. Importantly, other variables such as GP hyper-parameters $\boldsymbol{\theta}$ and inducing locations $\mathbf{Z}$ can be easily included in $\boldsymbol{\Psi}$ using suitable priors and incorporating them in our objective in Eq. (21).

4.3.1. Computational Cost & Prior Whitening

An interesting aspect of GPSSM models is that, despite their apparent Markovian nature, sampling at time T requires conditioning on all the previous $T-1$ points. This is due to the non-parametric coupled GP prior over the transition function, making inference in the full model $\mathcal{O}(T^3)$ in time. Sparse variational inference approaches, such as those based on inducing variable approximations, still require expectations over entire trajectories and, unlike standard supervised i.i.d settings, their time complexity is inherently dependent on T. Evaluation of the stochastic gradient Hamiltonian Monte Carlo (SGHMC) objective in Eq. (21) for sampling in our free-form variational dynamics (FFVD) algorithm is $\mathcal{O}(M^2T)$. this is the same cost as that attained for ELBO evaluation in variationally coupled dynamic trajectories (VCDT), which like our FFVD, models dependencies between state trajectories and inducing variables.

We may have a further comparison between our FFVD and the VCDT. Our SGHMC objective in Eq. (21) is very similar to the ELBO used in VCDT. The ELBO in VCDT requires expectations over $q(\mathbf{u})$) and $q(\mathbf{x}_t, \mathbf{x}_{t-1}|\mathbf{u})$. Given the factorization assumptions and the Gaussian constraints on these distributions, these expectations are estimated straightforwardly via Monte Carlo samples. The time complexity of evaluating Eq. (21) or ELBO in VCDT once (using one sample) is the same, i.e., $\mathcal{O}(M^3 + TM^2)$.

The overall time complexity of both algorithms, FFVD and VCDT, depends on (i) the number of samples (noting, again, that VCDT also requires samples from the approximate posterior to estimate the gradients of the ELBO) and (ii) the number of iterations (either the length of the SGHMC chain in FFVD or the number of epochs for gradient-based optimization in VCDT). As described in the appendix, our experimental setting followed closely that of the original VCDT paper, which used $S = 100$ samples for training and $S = 10^5$ for predictions. Similarly, we used $S = 100$ samples for FFVD. Remarkably, the number of iterations in our experiments for FFVD was 50 000 while for VCDT was 200 000 to achieve convergence. Furthermore, our analysis in the appendix shows that, in fact, our FFVD algorithm converges in less than 10,000 iterations.

As in previous work (see, e.g., Hensman et al., 2015), we have observed that whitening the prior over the inducing variables improves the performance of our algorithm. See details of our whitening procedure and the resulting unnormalized log posterior in Appendices D and E.1.

4.4. Smoothing and Predictive Distributions

We are interested in estimating the smoothing distribution $p(\mathbf{x}_{0:T} \,|\, \mathbf{y}_{1:T})$ and the predictive distribution $p(\mathbf{y}_{T+1:T'} \,|\, \mathbf{y}_{1:T})$ for $T' \geq T$. At the end of our SGHMC procedure, we have S samples from our joint approximate posterior $q(\mathbf{x}_{0:T}, \mathbf{u}) \approx p(\mathbf{x}_{0:T} \,|\, \mathbf{y}_{1:T})$, i.e., $\{\mathbf{x}_{0:T}^{(s)}, \mathbf{u}^{(s)}\}_{s=1}^{S}$ and, therefore, the smoothing distribution (i.e., the marginal posterior over $\mathbf{x}_{0:T}$) is readily available through this Monte Carlo approximation.

We can also make one-step-ahead predictions using our posterior samples. In particular, using Eqs. (2) and (17) we have that:

$$p(\mathbf{x}_t \,|\, \mathbf{x}_{t-1}, \mathbf{u}, \mathbf{Z}) = \mathcal{N}(\mathbf{x}_t; \mathbf{m}_{t-1} + \mathbf{A}_{t-1}(\mathbf{u} - \mathbf{m}_Z), \mathbf{B}_{t-1} + \mathbf{Q}), \quad (22)$$

$\forall t > T$. Thus, replacing the values of $\mathbf{A}_{t-1}, \mathbf{B}_{t-1}$ using Eqs. (19) and (20) we can make predictions for the next state using samples as

$$\begin{aligned} \mathbf{x}_t^{(s)} \,|\, \mathbf{x}_{t-1}^{(s)}, \mathbf{u}^{(s)} &\sim \mathcal{N}(\mathbf{x}_t; \boldsymbol{\mu}_t^{(s)}, \boldsymbol{\Sigma}_t^{(s)}) \quad \text{with} \\ \boldsymbol{\mu}_t^{(s)} &= \mathbf{m}_{t-1}^{(s)} + \mathbf{K}_{t-1,Z}^{(s)} \mathbf{K}_Z^{-1} (\mathbf{u}^{(s)} - \mathbf{m}_Z), \\ \boldsymbol{\Sigma}_t^{(s)} &= \mathbf{K}_{t-1}^{(s)} - \mathbf{K}_{t-1,Z}^{(s)} \mathbf{K}_Z^{-1} \mathbf{K}_{Z,t-1}^{(s)} + \mathbf{Q}. \end{aligned} \quad (23)$$

For a general likelihood model, we can sample the (noisy) targets using $\mathbf{y}_t^{(s)} \,|\, \mathbf{x}_t^{(s)} \sim p(\mathbf{y}_t \,|\, \mathbf{x}_t^{(s)}, \boldsymbol{\phi})$. In the case of a Gaussian conditional likelihood, as described in § 3, we can see that given samples from the latent state, the predictive distribution is a Gaussian

$$\mathbf{y}_t^{(s)} \,|\, \mathbf{x}_{t-1}^{(s)}, \mathbf{u}^{(s)} \sim \mathcal{N}(\mathbf{y}_t; \mathbf{C}\boldsymbol{\mu}_t^{(s)} + \mathbf{d}, \mathbf{C}\boldsymbol{\Sigma}_t^{(s)}\mathbf{C}^\top + \mathbf{R}). \quad (24)$$

where the samples of $\{\mathbf{x}_T^{(s)}, \mathbf{u}^{(s)}\}$ are readily available after running SGHMC on their joint space.

5. Collapsing Inducing Variables

So far we have described our method to obtain samples from the optimal variational posterior over the joint distribution of state trajectories and inducing variables using Eq. (21) and SGHMC. In this section we show that we can, in fact, integrate out the inducing variables $\mathbf{u}$ from our joint variational distribution and obtain the optimal marginal distribution for latent states $\mathbf{x}_{0:T}$. We start by retaking Eq. (21) and isolating the terms that depend on the inducing variables,

$$\begin{aligned} q^*(\mathbf{x}_{0:T}) &= \int_{\mathbf{u}} q^*(\mathbf{u}, \mathbf{x}_{0:T}) d\mathbf{u} \\ &= p(\mathbf{x}_0) \prod_{t=1}^{T} \left[p(\mathbf{y}_t \,|\, \mathbf{x}_t) \exp(-\frac{1}{2} \operatorname{Tr}(\mathbf{Q}^{-1}\mathbf{B}_{t-1})) \right] \\ &\quad \int_{\mathbf{u}} p(\mathbf{u} \,|\, \mathbf{Z}) \prod_{t=1}^{T} \mathcal{N}(\mathbf{x}_t; \boldsymbol{\mu}_{x_t}, \mathbf{Q}) \mathrm{d}\mathbf{u}, \end{aligned} \quad (25)$$

where we note that $\boldsymbol{\mu}_{x_t}$, as defined as in Eq. (18), depends on the inducing variables $\mathbf{u}$. With this, we can complete the square and identify the terms in the integral as products of Gaussian distributions, whose normalization constant is a Gaussian. Therefore, we have

$$q^*(\mathbf{x}_{0:T}) = p(\mathbf{x}_0) \prod_t \Big[p(\mathbf{y}_t|\mathbf{x}_t) \exp(-\frac{1}{2} \operatorname{Tr}(\mathbf{Q}^{-1}\mathbf{B}_{t-1})) \mathcal{N}(\mathbf{x}_t; \mathbf{m}_{t-1}, \mathbf{Q}) \Big] / \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{0}, \tilde{\boldsymbol{\Sigma}}_x), \quad (26)$$

where the Gaussian density in the denominator is determined by

$$\tilde{\mathbf{x}} = \sum_{t=1}^{T} \tilde{\mathbf{A}}_{t-1}^\top \mathbf{Q}^{-1}(\mathbf{x}_t - \mathbf{m}_{t-1}), \quad (27)$$

$$\tilde{\boldsymbol{\Sigma}}_x = \mathbf{I} + \sum_{t=1}^{T} \tilde{\mathbf{A}}_{t-1}^\top \mathbf{Q}^{-1} \tilde{\mathbf{A}}_{t-1}, \quad (28)$$

$$\tilde{\mathbf{A}}_{t-1} = \mathbf{K}_{t-1,Z}(\mathbf{L}_Z^\top)^{-1}, \quad (29)$$

where $\mathbf{L}_Z$ is the Cholesky decomposition of $\mathbf{K}_Z$, i.e., $\mathbf{K}_Z = \mathbf{L}_Z\mathbf{L}_Z^\top$. We note here that $\tilde{\mathbf{x}}$ is actually a projection of the cumulative uncorrelated inputs $\mathbf{Q}^{-1}(\mathbf{x}_t - \mathbf{m}_{t-1})$ via the projection matrix $\mathbf{L}_Z^{-1}\mathbf{K}_{Z,t-1}$ and, therefore, $\tilde{\mathbf{x}} \in \mathbb{R}^M$.

Furthermore, in this collapsed version, the terms unrelated to $\mathbf{u}$ are kept the same as those in the original optimal joint distribution in Eq. (21). The individual Gaussian transitions for the latent states are now $\mathcal{N}(\mathbf{x}_t; \mathbf{m}_{t-1}, \mathbf{Q})$ since $\mathbf{u}$ has been integrated out and we recall that $\mathbf{m}_{t-1} = m_f(\mathbf{x}_{t-1})$. The additional term, which is the inverse of a multivariate Gaussian distribution, records the cumulative projected trajectory with the corresponding projected variances. Maximizing $q^*(\mathbf{x}_{0:T})$ would tend to minimize this density function, which pushes the latent states away from $\mathbf{0}$ and also decreases the values of the $\mathbf{Q}$ regulated variance.

In the prediction scenario, we run SGHMC to obtain samples $\{\mathbf{x}_{0:T}^{(s)}\}$ from the marginal $q^*(\mathbf{x}_{0:T})$ and then use the closed-form expression for the conditional (Section G in the Supplementary Material) to obtain $\{\mathbf{u}^{(s)}|\mathbf{x}_{0:T}^{(s)}\}$.

5.1. Advantages of Collapsed Algorithm

Collapsing the inducing variables $\mathbf{u}$ will generally tend to improve the convergence of our algorithm, as we are required to sample from a significantly lower number of latent variables. The computational cost is similar to that of the uncollapsed algorithm since despite avoiding the computation of $\mathbf{K}_Z^{-1}$ in $p(\mathbf{u} \mid \mathbf{Z})$, we require a similar term, $\mathbf{L}_Z^{-1}$, in the evaluation of the projected Gaussian distributions.

It is important to emphasize one particular difference in our approach with respect to the closely related VCDT algorithm of Ialongo et al. (2019). As described before, Ialongo et al. (2019) also propose a coupled joint posterior between state trajectories and inducing variables. Their factorization is $q_{\text{VCDT}} = q(\mathbf{u})q(\mathbf{x}_{0:T} \mid \mathbf{u})$, and they impose additional Gaussian constraints on these densities. Our implicit assumed factorization is $q_{\text{FFVD}} = q(\mathbf{x}_{0:T})q(\mathbf{u} \mid \mathbf{x}_{0:T})$, which allows us to obtain the optimal variational distribution without imposing any additional parametric constraints, integrate out the inducing variables analytically and get a Monte Carlo approximation to the optimal marginal $q(\mathbf{x}_{0:T})$ via samples. We also provide an expression for the conditional $q(\mathbf{u} \mid \mathbf{x}_{0:T})$ in closed-form. Details can be found in Appendix G.

5.2. Particle Markov chain Monte Carlo (PMCMC)

It is clear that the latent states are constructed in a Markovian manner when the transition function $f(\cdot)$ is given, as the value of the current latent state is dependent on the previous latent state's value. Therefore, we can use PMCMC methods (Andrieu et al., 2010) to infer the posterior distribution of the Markov structured latent states $\mathbf{x}_{0:T}$. This Bayesian treatment might improve performance over SGHMC, as it incorporates the sequential nature of the problem into the sampling algorithm. The advantages of PMCMC over standard sequential Monte Carlo approaches have been documented previously, see. e.g., Andrieu et al. (2010). Here we note that Frigola et al. (2013) also proposed a PMCMC treatment for $\mathbf{x}_{0:T}$. However, their algorithm is very different from ours as it is based on the fully-independent conditional approximation (see, e.g., Quinonero-Candela & Rasmussen, 2005). More details can be found in Appendix K.

6. Related Work

We have already described the main differences between our method and closely-related approaches throughout the paper, e.g., in §§ 1, 4 and 5. We refer the reader to Appendix K for more details. Other works have considered the GPSSM in partially observable unstable settings (Curi et al., 2020) or combined variational inference with the Laplace approximation (Lindinger et al., 2022). With regards to scalable GPs, we note they have been the subject of much research effort in machine learning, with extensions to more general frameworks such as compositional models (see, e.g., Wilson & Nickisch, 2015; Salimbeni & Deisenroth, 2017; Yu et al., 2019; Cutajar et al., 2017; Havasi et al., 2018; Rossi et al., 2021). Our approach can be seen as a generalization of the fully-Bayesian supervised learning method proposed by Rossi et al. (2021) to state-space models, where we have included the non-trivial component of GP-transition dynamics and have proposed a collapsed optimal variational distribution for state trajectories.

Other models for GPs on sequential data have been proposed, see, for example, Frigola (2015) for an excellent overview. Of interest here is the state-space model view

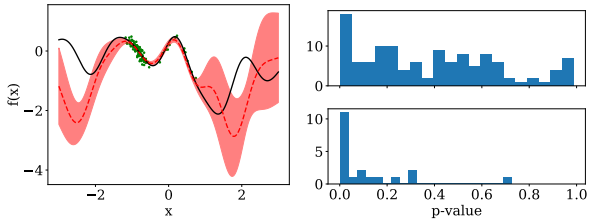

Figure 1. Results on synthetic data. *Left:* Observations shown as green dots, Ground truth as a solid black line, and FFVD's mean fitting as a dashed red line with one standard deviation error bars. *Right:* Histograms of p-values for the hypothesis test that each marginal posterior over states $\{x_t\}$ (top) and inducing variables $\{u_i\}$ (bottom) is generated from a Normal distribution.

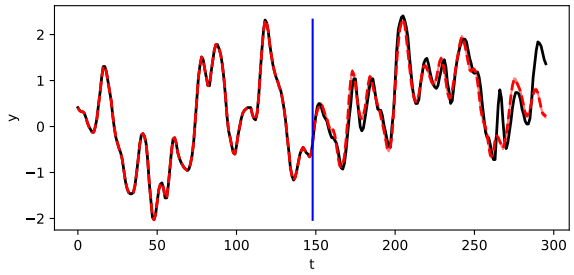

Figure 2. Training ($t <= 150$) and test performance ($t >= 150$) on the Furnace dataset. The black solid line is the underlying ground truth signal and the dashed red line is FFVD's mean prediction. The blue solid line indicates the training/test split. An underlying $d_x = 4$-dimensional latent stated was used.

of GPs that for time-series data with $d_x = 1$ and Markovian covariance functions can provide exact inference in linear time $\mathcal{O}(T)$ (Solin, 2016). This has been extended to non-Gaussian likelihood models and made more efficient using several computational primitives (Nickisch et al., 2018). More recently, there has been some work on using the signature kernel (Toth & Oberhauser, 2020; Salvi et al., 2021) within GP models. In particular, Lemercier et al. (2021) generalize variational orthogonal features (Burt et al., 2020; Hensman et al., 2017) to the sequential case, constructing inducing variables associated with the signature kernel that yield a variational inference algorithm that does not require any matrix inversion.

7. Experiments

We evaluate our FFVD method on synthetic data and on six real-world system identification benchmarks (Ialongo et al., 2019; Doerr et al., 2018), comparing it with VGPSSM (Frigola et al., 2014), PRSSM (Doerr et al., 2018), VCDT (Ialongo et al., 2019), and use a LSTM network (Hochreiter & Schmidhuber, 1997) as a baseline non-GP based model. All experiment details can be found in Appendix I.

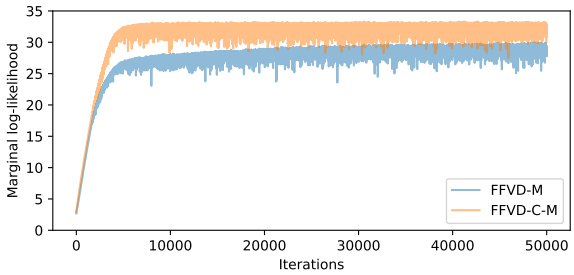

Figure 3. Traceplot of the training log-likelihood when using SGHMC (FFVD-M) and when collapsing the inducing variables FFVD-C-M. Collapsing generally improved convergence.

7.1. Synthetic Data

We generate data from a sparse GPSSM with a squared exponential covariance function. Our goal here is to investigate the properties of our FFVD algorithm. Therefore, we fix the value of all parameters to their ground-truth values except for the latent states $\mathbf{x}_{0:T}$ and the inducing variables $\mathbf{u}$. The left panel of Fig. 1 illustrates that FFVD effectively learns the intricate transition function with two modes within regions containing latent states. As the number of latent states increases, FFVD more accurately approximates the true function. A lack of fit in regions without latent states is to be expected. Having a good fit, we are now interested in knowing whether the true posterior (as estimated by our algorithm) is close to a Gaussian distribution. The right panel in Fig. 1 indicates that more than 10% of the latent states $\mathbf{x}_{0:T}$ and more than 50% of the inducing variables $\mathbf{u}$ do not provide enough support for the hypothesis that their marginal distributions follow a Normal distribution (see details in Appendix I). This brings into question the strong parametric assumptions over the variational posterior made by previous work (e.g., Ialongo et al., 2019; Doerr et al., 2018).

7.2. Real-World Data

Here we evaluate the different methods using six system identification benchmarks with a latent state dimension $d_x = 4$, as used by Ialongo et al. (2019); Doerr et al. (2018). See more details in Appendix I. The prediction performance is shown in Table 2 and Table 3, where we see that FFVD attains the lowest RMSE values in four out of the six benchmarks (in bold). Furthermore, for *Actuator* and *Dryer*, FFVD-C-M ranks second among all the algorithms (underlined). The performance of VGPSSM and PRSSM is usually worse than others, which is likely due to their strong mean-field and parametric assumptions, respectively. LSTM obtains good performance on three datasets, although its deterministic neural network structure is different from our random function setting. The performance of VCDT is

Table 2. Mean RMSE values $\pm$ one standard deviations on the real-world system identification benchmarks. Our method, FFVD, when using SGHMC (FFVD-M); the collapsed version (FFVD-C-M); and when using PMCMC (FFVD-P).

Methods	Actuator	Ballbeam	Drive	Dryer	Flutter	Furnace
LSTM	0.586 ± 0.411	0.027 ± 0.023	0.537 ± 0.108	0.115 ± 0.029	0.912 ± 0.562	1.261 ± 0.610
VGPSSM	0.580 ± 0.274	0.073 ± 0.011	0.722 ± 0.087	0.241 ± 0.023	1.482 ± 0.218	1.115 ± 0.358
PRSSM	0.497 ± 0.381	0.059 ± 0.013	0.813 ± 0.101	$\mathbf{0.017} \pm 0.042$	1.371 ± 0.156	1.243 ± 0.407
VCDT	$\mathbf{0.239} \pm 0.040$	0.011 ± 0.002	0.585 ± 0.017	0.142 ± 0.003	1.782 ± 0.324	1.166 ± 0.011
FFVD-M	0.358 ± 0.242	0.019 ± 0.018	0.673 ± 0.207	0.205 ± 0.313	$\mathbf{0.280} \pm 0.193$	0.571 ± 0.185
FFVD-C-M	$\underline{0.259} \pm 0.209$	$\mathbf{0.009} \pm 0.011$	0.775 ± 1.615	$\underline{0.065} \pm 0.112$	0.663 ± 0.189	$\mathbf{0.548} \pm 0.051$
FFVD-P	0.388 ± 0.087	0.199 ± 0.045	$\mathbf{0.342} \pm 0.057$	0.317 ± 0.050	0.562 ± 0.088	0.669 ± 0.174

Table 3. Negative Log-Likelihood values on the real-world system identification benchmarks.

Methods	Actuator	Ballbeam	Drive	Dryer	Flutter	Furnace
VCDT	-0.36 ± 0.02	-0.65 ± 0.01	1.23 ± 0.01	$\mathbf{-0.02} \pm 0.01$	$\mathbf{6.13} \pm 0.48$	$\mathbf{7.49} \pm 0.07$
FFVD-M	$\mathbf{-0.03} \pm 0.10$	$\mathbf{0.09} \pm 0.05$	$\mathbf{1.66} \pm 0.01$	-0.08 ± 0.09	0.48 ± 0.33	-0.43 ± 0.05

Table 4. Mean running time values on Furnace

Methods	Iterations	Furnace
VCDT	10	26.96
	100	269.71
	500	1613.25
	1 000	3374.78
FFVD-M	10	3.19
	100	26.33
	500	121.42
	1 000	238.68

the closest to our FFVD methods, as it models a coupled $q(\mathbf{x}_{0:T}, \mathbf{u})$.

In addition to this quantitative evaluation, we can see a qualitative illustration of using our algorithm for predicting the training and test (future) observations in Fig. 2. This is an example of good generalization, although it is (of course) not consistent across all problems, given the limited training data. Finally, we analyze the convergence of our algorithm in Fig. 3, where we see that FFVD-C-M uses less iterations ($\sim 8\,000$ iterations) than FFVD-M ($\sim 40\,000$ iterations) to achieve similar performance. This confirms the benefits of collapsing the inducing variables and sampling only on the lower-dimensional space of state trajectories. We can see these analyses for all benchmarks in Appendices J.1 and J.2.

We also have executed experiments to illustrate the advantages of our approach when considering running time regarding to different number of iterations in Table 4. We can clearly see that our approach is 10 times faster than VCDT. These time results were done using a Macbook Pro 2021 with 16GB in memory, M1 chip, and 8 cores. It is noted that caution must be taken with interpreting these results, as they depend on implementation specifics, computer architectures along with other practical details.

8. Limitations

Our approach is an approximation. Indeed, we did not realize our paper came across as claiming our posterior was the true posterior or exact in any particular limiting case. Such an approach for defining the conditional posterior in terms of the conditional prior is customary in variational methods for GP models since the seminal work of Titsias (2009). Sampling from the true full (non-sparse) posterior is not scalable due to its cubic cost on the number of observations.

9. Conclusion

We have presented FFVD, a new algorithm for inference in GPSSMs based on variational inference under the inducing-variable formalism. Unlike previous approaches, FFVD does not make any independence or parametric assumptions on the joint variational posterior over state trajectories and inducing variables $q(\mathbf{x}_{0:T}, \mathbf{u})$ and, instead, represents the posterior via samples form the optimal variational distribution. Furthermore, we have shown that our formulation allows us to collapse the inducing variables and carry out inference in the lower dimensional space of state trajectories. Given samples from trajectories, we can make predictions using the optimal conditional $q(\mathbf{u} \,|\, \mathbf{x}_{0:T})$, for which we have derived a closed-form expression. Future work will investigate scalable multi-output GPs in order to deal with very high-dimensional state representations in a flexible and efficient way.

References

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

Bauer, M., van der Wilk, M., and Rasmussen, C. E. Understanding probabilistic sparse gaussian process approximations. *Advances in neural information processing systems*, 29, 2016.

Brown, E. N., Frank, L. M., Tang, D., Quirk, M. C., and Wilson, M. A. A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells. *Journal of Neuroscience*, 18(18):7411–7425, 1998.

Burt, D. R., Rasmussen, C. E., and van der Wilk, M. Variational orthogonal features. *arXiv preprint arXiv:2006.13170*, 2020.

Chen, T., Fox, E., and Guestrin, C. Stochastic gradient Hamiltonian Monte Carlo. In *ICML*, pp. 1683–1691. PMLR, 2014.

Curi, S., Melchior, S., Berkenkamp, F., and Krause, A. Structured variational inference in partially observable unstable Gaussian process state space models. In Bayen, A. M., Jadbabaie, A., Pappas, G., Parrilo, P. A., Recht, B., Tomlin, C., and Zeilinger, M. (eds.), *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, volume 120 of *Proceedings of Machine Learning Research*, pp. 147–157. PMLR, 10–11 Jun 2020.

Cutajar, K., Bonilla, E. V., Michiardi, P., and Filippone, M. Random feature expansions for deep Gaussian processes. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 884–893. PMLR, 06–11 Aug 2017.

D'agostino, R. and Pearson, E. S. Tests for departure from normality. *Biometrika*, 60(3):613–622, 1973.

Doerr, A., Daniel, C., Schiegg, M., Duy, N.-T., Schaal, S., Toussaint, M., and Sebastian, T. Probabilistic recurrent state-space models. In *ICML*, pp. 1280–1289, 2018.

Eleftheriadis, S., Nicholson, T., Deisenroth, M., and Hensman, J. Identification of Gaussian process state space models. *Advances in neural information processing systems*, 30, 2017.

Frigola, R. *Bayesian time series learning with Gaussian processes*. PhD thesis, University of Cambridge, 2015.

Frigola, R., Lindsten, F., Schön, T. B., and Rasmussen, C. E. Bayesian inference and learning in Gaussian process state-space models with particle MCMC. In *NIPS*, pp. 3156–3164, 2013.

Frigola, R., Chen, Y., and Rasmussen, C. E. Variational Gaussian process state-space models. In *NIPS*, pp. 3680–3688, 2014.

Havasi, M., Hernández-Lobato, J. M., and Murillo-Fuentes, J. J. Inference in deep Gaussian processes using stochastic gradient Hamiltonian Monte Carlo. *NeurIPS*, 31, 2018.

Hensman, J., Fusi, N., and Lawrence, N. D. Gaussian processes for big data. In *Uncertainty in Artificial Intelligence*. AUAI Press, 2013.

Hensman, J., Matthews, A. G., Filippone, M., and Ghahramani, Z. MCMC for variationally sparse Gaussian processes. In *NIPS*, pp. 1648–1656, 2015.

Hensman, J., Durrande, N., Solin, A., et al. Variational Fourier features for Gaussian processes. *J. Mach. Learn. Res.*, 18(1):5537–5588, 2017.

Hernandez, E., Martin, F., and Valero, F. State-space modeling for atmospheric pollution. *Journal of Applied Meteorology and Climatology*, 30(6):793–811, 1991.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Ialongo, A. D., Van Der Wilk, M., Hensman, J., and Rasmussen, C. E. Overcoming mean-field approximations in recurrent Gaussian process models. In *ICML*, pp. 2931–2940, 2019.

Kalman, R. E. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82 (1):35–45, 1960. doi: 10.1115/1.3662552.

Katzfuss, M., Stroud, J. R., and Wikle, C. K. Understanding the Ensemble Kalman Filter. *The American Statistician*, 70(4):350–357, 2016. doi: 10.1080/00031305.2016.1141709.

Kitagawa, G. Non-Gaussian state—space modeling of nonstationary time series. *Journal of the American statistical association*, 82(400):1032–1041, 1987.

Lemercier, M., Salvi, C., Cass, T., Bonilla, E. V., Damoulas, T., and Lyons, T. J. SigGPDE: Scaling sparse Gaussian processes on sequential data. In *International Conference on Machine Learning*, pp. 6233–6242. PMLR, 2021.

Lindinger, J., Rakitsch, B., and Lippert, C. Laplace approximated Gaussian process state-space models. In *The 38th Conference on Uncertainty in Artificial Intelligence*, 2022.

Marmin, S. and Filippone, M. Deep gaussian processes for calibration of computer models (with discussion). *Bayesian Analysis*, 17(4):1301–1350, 2022.

Murphy, K. P. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2023. URL http://probml.github.io/book2.

Nickisch, H., Solin, A., and Grigorevskiy, A. State space Gaussian processes with non-Gaussian likelihood. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3789–3798. PMLR, 10–15 Jul 2018.

Ogata, K. et al. *Modern control engineering*, volume 5. Prentice hall Upper Saddle River, NJ, 2010.

Quinonero-Candela, J. and Rasmussen, C. E. A unifying view of sparse approximate Gaussian process regression. *The Journal of Machine Learning Research*, 6: 1939–1959, 2005.

Rossi, S., Heinonen, M., Bonilla, E., Shen, Z., and Filippone, M. Sparse Gaussian processes revisited: Bayesian approaches to inducing-variable approximations. In *AISTATS*, pp. 1837–1845, 2021.

Salimbeni, H. and Deisenroth, M. Doubly stochastic variational inference for deep Gaussian processes. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Salvi, C., Cass, T., Foster, J., Lyons, T., and Yang, W. The signature kernel is the solution of a goursat PDE. *SIAM Journal on Mathematics of Data Science*, 3(3):873–899, 2021.

Sauer, A., Cooper, A., and Gramacy, R. B. Vecchia-approximated deep Gaussian processes for computer experiments. *Journal of Computational and Graphical Statistics*, 0(0):1–14, 2022.

Solin, A. *Stochastic differential equation methods for spatio-temporal Gaussian process regression*. PhD thesis, Aalto University, 2016.

Titsias, M. Variational learning of inducing variables in sparse Gaussian processes. In *AISTATS*, pp. 567–574, 2009.

Toth, C. and Oberhauser, H. Bayesian learning from sequential data using Gaussian processes with signature covariances. In *International Conference on Machine Learning*, pp. 9548–9560. PMLR, 2020.

Tsay, R. S. *Analysis of financial time series*. John wiley & sons, 2005.

Williams, C. K. and Rasmussen, C. E. *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006.

Wilson, A. and Nickisch, H. Kernel interpolation for scalable structured Gaussian processes (KISS-GP). In Bach, F. and Blei, D. (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1775–1784, Lille, France, 07–09 Jul 2015. PMLR.

Yu, H., Chen, Y., Low, B. K. H., Jaillet, P., and Dai, Z. Implicit Posterior Variational Inference for Deep Gaussian Processes. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 14475–14486. Curran Associates, Inc., 2019.

A. Basic Results on Gaussian Distributions

A.1. Conditional Gaussian Distribution

Assuming the joint Gaussian distribution $p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ over the random vector $\mathbf{x}$ such that:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_a \\ \mathbf{x}_b \end{bmatrix}, \quad \boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_a \\ \boldsymbol{\mu}_b \end{bmatrix}, \quad \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{aa} & \boldsymbol{\Sigma}_{ab} \\ \boldsymbol{\Sigma}_{ba} & \boldsymbol{\Sigma}_{bb}, \end{bmatrix} \tag{30}$$

where $\boldsymbol{\Sigma}_{ba} = \boldsymbol{\Sigma}_{ab}^T$, we have that the conditional distributions are given by:

$$p(\mathbf{x}_b \,|\, \mathbf{x}_a) = \mathcal{N}(\mathbf{x}_b; \boldsymbol{\mu}_b + \boldsymbol{\Sigma}_{ba}\boldsymbol{\Sigma}_{aa}^{-1}(\mathbf{x}_a - \boldsymbol{\mu}_a), \boldsymbol{\Sigma}_{bb} - \boldsymbol{\Sigma}_{ba}\boldsymbol{\Sigma}_{aa}^{-1}\boldsymbol{\Sigma}_{ab}). \tag{31}$$

A.2. Expectation over log of Normal Distribution

With an approximate marginal posterior $q(\mathbf{f}_\star)$ and a Normal distribution $p(\mathbf{y} \,|\, \mathbf{f}_\star)$ of the form:

$$q(\mathbf{f}_\star) = \mathcal{N}(\mathbf{f}_\star; \boldsymbol{\mu}_\star, \boldsymbol{\Sigma}_\star), \tag{32}$$

$$p(\mathbf{y} \,|\, \mathbf{f}_\star) = \mathcal{N}(\mathbf{y}; \mathbf{f}_\star, \boldsymbol{\Sigma}_y), \tag{33}$$

we can compute

$$\mathbb{E}_{q(\mathbf{f}_\star)} \log p(\mathbf{y} \,|\, \mathbf{f}_\star) = \log \mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_\star, \boldsymbol{\Sigma}_y) + \mathrm{Tr}(\boldsymbol{\Sigma}_y^{-1}\boldsymbol{\Sigma}_\star). \tag{34}$$

B. Posterior Marginal in Sparse GP Models

In variational sparse GP models we usually have the joint Gaussian model $p(\mathbf{f}_\star, \mathbf{u})$ and a posterior distribution over $q(\mathbf{u})$ with

$$p(\mathbf{f}_\star) = \mathcal{N}(\mathbf{m}_\star, \mathbf{K}_\star), \tag{35}$$

$$p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \mathbf{m}_Z, \mathbf{K}_Z), \tag{36}$$

$$\mathbb{C}\mathrm{ov}(\mathbf{f}_\star, \mathbf{u}) = \mathbf{K}_{\star,Z}, \tag{37}$$

$$q(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \boldsymbol{\mu}_\mathbf{u}, \boldsymbol{\Sigma}_\mathbf{u}), \tag{38}$$

$$q(\mathbf{f}_\star, \mathbf{u}) := q(\mathbf{u})p(\mathbf{f}_\star \,|\, \mathbf{u}), \tag{39}$$

and we wish to compute:

$$q(\mathbf{f}_\star) = \int q(\mathbf{f}_\star, \mathbf{u}) \, \mathrm{d}\mathbf{u}, \tag{40}$$

where we have omitted conditional dependencies on $\mathbf{x}_\star$ and $\mathbf{Z}$ for simplicity in the notation. For example this is necessary to compute the expectations over the conditional log likelihood term during inferencce or to make predictions on a new test point. Using Eq. (31) we obtain:

$$p(\mathbf{f}_\star \,|\, \mathbf{u}) = \mathcal{N}(\mathbf{f}_\star; \mathbf{m}_\star + \mathbf{K}_{\star,Z}\mathbf{K}_Z^{-1}(\mathbf{u} - \mathbf{m}_Z), \mathbf{K}_\star - \mathbf{K}_{\star,Z}\mathbf{K}_Z^{-1}\mathbf{K}_{Z,\star}). \tag{41}$$

In order to integrate out over $q(\mathbf{u})$ we know that the result is a Gaussian with mean and covariances obtained from the following linear transformation of $\mathbf{u}$:

$$\mathbf{f}_\star = \mathbf{m}_\star + \mathbf{K}_{\star,Z}\mathbf{K}_Z^{-1}(\mathbf{u} - \mathbf{m}_Z) + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_\star - \mathbf{K}_{\star,Z}\mathbf{K}_Z^{-1}\mathbf{K}_{Z,\star}). \tag{42}$$

Thus, we obtain:

$$q(\mathbf{f}_\star) = \mathcal{N}\left(\mathbf{f}_\star; \mathbf{m}_\star + \mathbf{K}_{\star,Z}\mathbf{K}_Z^{-1}(\boldsymbol{\mu}_\boldsymbol{\mu} - \mathbf{m}_Z), \mathbf{K}_{\star,Z}\mathbf{K}_Z^{-1}\boldsymbol{\Sigma}_\mathbf{u}\mathbf{K}_Z^{-1}\mathbf{K}_{Z,\star} + \mathbf{K}_\star - \mathbf{K}_{\star,Z}\mathbf{K}_Z^{-1}\mathbf{K}_{Z,\star}\right) \tag{43}$$

$$= \mathcal{N}\left(\mathbf{f}_\star; \mathbf{m}_\star + \mathbf{K}_{\star,Z}\mathbf{K}_Z^{-1}(\boldsymbol{\mu}_\boldsymbol{\mu} - \mathbf{m}_Z), \mathbf{K}_\star + \mathbf{K}_{\star,Z}\mathbf{K}_Z^{-1}\left(\boldsymbol{\Sigma}_\mathbf{u} - \mathbf{K}_Z\right)\mathbf{K}_Z^{-1}\mathbf{K}_{Z,\star}\right). \tag{44}$$

C. Graphical Model

Fig. 4 illustrates GPSSM's graphical model.

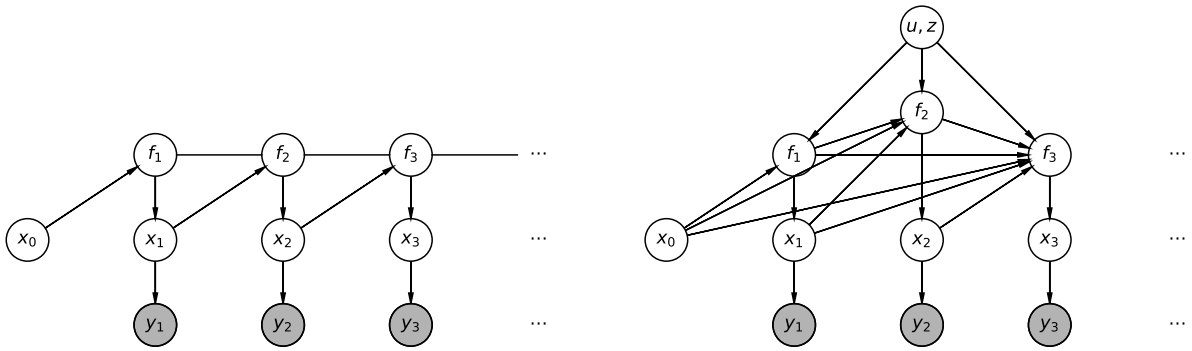

Figure 4. Generative process of the GPSSM (left panel) and the sparse GPSSM (right panel).

D. Details of Prior Reparameterization: Whitening Prior over Inducing Variables

Here we show a more detailed derivation of the prior reparameterization and, consequently, the new form of the required conditional distribution. We know that our prior over the inducing variables is $p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \mathbf{m}_Z, \mathbf{K}_Z)$ and that we use the new whitened prior $\mathbf{v} = \mathbf{L}_Z^{-1}(\mathbf{u} - \mathbf{m}_Z)$ with $\mathbf{L}_Z\mathbf{L}_Z^T = \mathbf{K}_Z$ and $p(\mathbf{v}) = \mathcal{N}(\mathbf{v}; \mathbf{0}, \mathbf{I}_M)$. We also have that the marginal prior over $\mathbf{f}_\star$ is $p(\mathbf{f}_\star) = \mathcal{N}(\mathbf{f}_\star; \mathbf{m}_\star, \mathbf{K}_\star)$. It follows that $p(\mathbf{f}, \mathbf{v})$ is a Gaussian with cross-covariance:

$$\begin{aligned}
\mathbb{C}\text{ov}(\mathbf{f}_\star, \mathbf{v}) &= \mathbb{E}[(\mathbf{f}_\star - \mathbf{m}_\star)(\mathbf{L}_Z^{-1}(\mathbf{u} - \mathbf{m}_Z))^T] && (45)\\
&= \underbrace{\mathbb{E}[(\mathbf{f}_\star - \mathbf{m}_\star)(\mathbf{u} - \mathbf{m}_Z)^T]}_{\mathbb{C}\text{ov}(\mathbf{f}_\star, \mathbf{u})}(\mathbf{L}_Z^{-1})^T && (46)\\
&= \mathbf{K}_{\star,Z}(\mathbf{L}_Z^{-1})^T. && (47)
\end{aligned}$$

Hence, using Eq. (31), we have that:

$$\begin{aligned}
p(\mathbf{f}_\star \,|\, , \mathbf{x}_\star, \mathbf{v}, \mathbf{Z}) &= \mathcal{N}(\mathbf{f}_\star; \mathbf{m}_\star + \mathbf{K}_{\star,Z}(\mathbf{L}_Z^{-1})^T\mathbf{v}, \mathbf{K}_\star - \mathbf{K}_{\star,Z}(\mathbf{L}_Z^{-1})^T\mathbf{L}_Z^{-1}\mathbf{K}_{Z,t}), && (48)\\
p(\mathbf{f}_\star \,|\, , \mathbf{x}_\star, \mathbf{v}, \mathbf{Z}) &= \mathcal{N}(\mathbf{f}_\star; \mathbf{m}_\star + \mathbf{K}_{\star,Z}(\mathbf{L}_Z^{-1})^T\mathbf{v}, \mathbf{K}_\star - \mathbf{K}_{\star,Z}\mathbf{K}_Z^{-1}\mathbf{K}_{Z,\star}). && (49)
\end{aligned}$$

Alternatively, we can simply obtain this result by replacing $\mathbf{u}$ with $\mathbf{u} = \mathbf{m}_Z + \mathbf{L}_Z\mathbf{v}$ in the conditional distribution. Here we note that this is *consistent* with the GP definition in that if we were to integrate out the variables $\mathbf{v}$ from the joint model $p(\mathbf{f}_\star, \mathbf{v})$ we would obtain the exact GP prior $p(\mathbf{f}_\star)$.

D.1. Posterior Marginal

With the whitened join model $p(\mathbf{f}_\star, \mathbf{v})$, Gaussian approximate posterior $q(\mathbf{v})$ and a variationally sparse GP model we have:

$$\begin{aligned}
p(\mathbf{f}_\star \,|\, \mathbf{v}) &= \mathcal{N}(\mathbf{f}_\star; \mathbf{m}_\star + \mathbf{K}_{\star,Z}(\mathbf{L}_Z^{-1})^T\mathbf{v}, \mathbf{K}_\star - \mathbf{K}_{\star,Z}\mathbf{K}_Z^{-1}\mathbf{K}_{Z,\star}), && (50)\\
q(\mathbf{v}) &= \mathcal{N}(\mathbf{v}; \boldsymbol{\mu}_\mathbf{v}, \boldsymbol{\Sigma}_\mathbf{v}), && (51)\\
q(\mathbf{f}_\star, \mathbf{v}) &:= q(\mathbf{v})p(\mathbf{f}_\star \,|\, \mathbf{v}). && (52)
\end{aligned}$$

It is easy to show that the posterior marginal $q(\mathbf{f}_\star) = \int q(\mathbf{f}_\star, \mathbf{v})\,\mathrm{d}\mathbf{v}$ is:

$$q_v(\mathbf{f}_\star) = \mathcal{N}(\mathbf{f}_\star; \mathbf{m}_\star + \mathbf{K}_{\star,Z}(\mathbf{L}_Z^{-1})^T\boldsymbol{\mu}_\mathbf{v}, \mathbf{K}_\star + \mathbf{K}_{\star,Z}(\mathbf{L}_Z^{-1})^T(\boldsymbol{\Sigma}_\mathbf{v} - \mathbf{I}_M)\mathbf{L}_Z^{-1}\mathbf{K}_{Z,\star}). \quad (53)$$

E. Free-Form Posterior Estimation By Bounding the Log Marginal Likelihood Directly

Computing the log marginal:

$$\log p(\mathbf{y}_{0:T-1}, \mathbf{x}_{0:T-1}, \mathbf{u} \,|\, \mathbf{Z}) = \log \left[p(\mathbf{u} \,|\, \mathbf{Z})p(\mathbf{x}_0)p(\mathbf{y}_0 \,|\, \mathbf{x}_0) \prod_{t=1}^{T-1} p(\mathbf{y}_t \,|\, \mathbf{x}_t) \right] + \mathcal{L}(\mathbf{x}_{0:T-1}, \mathbf{u}, \mathbf{Z}), \quad (54)$$

where

$$\mathcal{L}(\mathbf{x}_{0:T-1}, \mathbf{u}, \mathbf{Z}) := \left[\log \int_{\mathbf{f}_{1:T-1}} \prod_{t=1}^{T-1} p(\mathbf{x}_t \,|\, \mathbf{f}_t) p(\mathbf{f}_t \,|\, \mathbf{f}_{1:t-1}, \mathbf{x}_{0:t-1}, \mathbf{u}, \mathbf{Z}) \, \mathrm{d}\mathbf{f}_{1:T-1} \right], \tag{55}$$

which we note is a log of an expectation, which we can bounded using Jensen's inequality:

$$\mathcal{L}(\mathbf{x}_{0:T-1}, \mathbf{u}, \mathbf{Z}) \geq \mathbb{E}_{\prod_{t=1}^{T-1} p(\mathbf{f}_t \,|\, \mathbf{f}_{1:t-1}, \mathbf{x}_{0:t-1}, \mathbf{u}, \mathbf{Z})} \log \prod_{t=1}^{T-1} p(\mathbf{x}_t \,|\, \mathbf{f}_t) \tag{56}$$

$$= \sum_{t=1}^{T-1} \mathbb{E}_{p(\mathbf{f}_t \,|\, \mathbf{x}_{t-1}, \mathbf{u}, \mathbf{Z})} \log p(\mathbf{x}_t \,|\, \mathbf{f}_t), \tag{57}$$

where $p(\mathbf{f}_t \,|\, \mathbf{x}_{t-1}, \mathbf{u}, \mathbf{Z})$ is the GP predictive distribution over $\mathbf{f}_t$ at $\mathbf{x}_{t-1}$ given the (pseudo observations) inducing variables $\mathbf{u}$ at inducing inputs $\mathbf{Z}$:

$$p(\mathbf{f}_t \,|\, \mathbf{x}_{t-1}, \mathbf{u}, \mathbf{Z}) = \mathcal{N}(\mathbf{f}_t; \mathbf{A}_{t-1}\mathbf{u}, \mathbf{B}_{t-1}) \quad \text{with} \tag{58}$$

$$\mathbf{A}_{t-1} = \mathbf{K}_{t-1,Z}\mathbf{K}_Z^{-1} \tag{59}$$

$$\mathbf{B}_{t-1} = \mathbf{K}_{t-1} - \mathbf{K}_{t-1,Z}\mathbf{K}_Z^{-1}\mathbf{K}_{Z,t-1}. \tag{60}$$

Hence, the expectations in Eq. (57) can be computed in closed-form:

$$\mathbb{E}_{p(\mathbf{f}_t \,|\, \mathbf{x}_{t-1}, \mathbf{u}, \mathbf{Z})} \log p(\mathbf{x}_t \,|\, \mathbf{f}_t) = \log \mathcal{N}(\mathbf{x}_t; \mathbf{A}_{t-1}\mathbf{u}, \mathbf{Q}) - \frac{1}{2} \operatorname{Tr}(\mathbf{Q}^{-1}\mathbf{B}_{t-1}). \tag{61}$$

Thus, we have that:

$$\log p(\mathbf{y}_{0:T-1}, \mathbf{x}_{0:T-1}, \mathbf{u} \,|\, \mathbf{Z}) \geq \log \left[p(\mathbf{u} \,|\, \mathbf{Z}) p(\mathbf{x}_0) p(\mathbf{y}_0 \,|\, \mathbf{x}_0) \right] + \sum_{t=1}^{T-1} \left[\log p(\mathbf{y}_t \,|\, \mathbf{x}_t) + \log \mathcal{N}(\mathbf{x}_t; \mathbf{A}_{t-1}\mathbf{u}, \mathbf{Q}) - \frac{1}{2} \operatorname{Tr}(\mathbf{Q}^{-1}\mathbf{B}_{t-1}) \right]. \tag{62}$$

Then setting the latent variables $\mathbf{\Psi} := \{\mathbf{u}, \mathbf{x}_{0:T-1}\}$, we can obtain the approximate log unnormalized posterior:

$$\log \tilde{q}(\mathbf{\Psi}) := \log p(\mathbf{u} \,|\, \mathbf{Z}) + \log p(\mathbf{x}_0) + \log p(\mathbf{y}_0 \,|\, \mathbf{x}_0) + \sum_{t=1}^{T-1} \left[\log p(\mathbf{y}_t \,|\, \mathbf{x}_t) + \log \mathcal{N}(\mathbf{x}_t; \mathbf{A}_{t-1}\mathbf{u}, \mathbf{Q}) - \frac{1}{2} \operatorname{Tr}(\mathbf{Q}^{-1}\mathbf{B}_{t-1}) \right]. \tag{63}$$

where $\tilde{q}(\mathbf{\Psi}) \propto q(\mathbf{\Psi}) \approx p(\mathbf{\Psi} \,|\, \mathbf{y}_{0:T-1})$. Thus, we can draw samples from our approximate posterior using SGHMC and the energy function $U(\mathbf{\Psi}) = -\log p(\mathbf{\Psi}, \mathbf{y}_{0:T-1}) = -\log p(\mathbf{\Psi} \,|\, \mathbf{y}_{0:T-1}) + C \approx -\log \tilde{q}(\mathbf{\Psi}) + C$.

E.1. Whitened Version

In the whitened version, $\mathbf{v} = \mathbf{L}_Z^{-1}(\mathbf{u} - \mathbf{m}_Z)$ or $\mathbf{u} = \mathbf{m}_Z + \mathbf{L}_Z\mathbf{v}$ and the effective prior is

$$p(\mathbf{v}) = \mathcal{N}(\mathbf{v}; \mathbf{0}, \mathbf{I}). \tag{64}$$

Letting $\tilde{\mathbf{u}} = (\mathbf{u}, \mathbf{Z}, \theta)$, $\tilde{\mathbf{v}} = (\mathbf{v}, \mathbf{Z}, \theta)$, the determinant of the Jacobian can be calculated as:

$$\left|\frac{\partial \tilde{\mathbf{u}}}{\partial \tilde{\mathbf{v}}}\right| = \begin{vmatrix} \frac{\partial(\mathbf{m}_Z + \mathbf{L}_Z\mathbf{v})}{\partial \mathbf{v}} & \frac{\partial(\mathbf{m}_Z + \mathbf{L}_Z\mathbf{v})}{\partial \mathbf{Z}} & \frac{\partial(\mathbf{m}_Z + \mathbf{L}_Z\mathbf{v})}{\partial \theta} \\ \frac{\partial \mathbf{Z}}{\partial \mathbf{v}} & \frac{\partial \mathbf{Z}}{\partial \mathbf{Z}} & \frac{\partial \mathbf{Z}}{\partial \theta} \\ \frac{\partial \theta}{\partial \mathbf{v}} & \frac{\partial \theta}{\partial \mathbf{Z}} & \frac{\partial \theta}{\partial \theta} \end{vmatrix} = \begin{vmatrix} \mathbf{L}_Z & \frac{\partial(\mathbf{m}_Z + \mathbf{L}_Z\mathbf{v})}{\partial \mathbf{Z}} & \frac{\partial(\mathbf{m}_Z + \mathbf{L}_Z\mathbf{v})}{\partial \theta} \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I} \end{vmatrix} = \begin{vmatrix} \mathbf{L}_Z & 0 & 0 \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I} \end{vmatrix} = |\mathbf{L}_Z| = |\mathbf{K}_Z|^{\frac{1}{2}} \tag{65}$$

Thus, we can re-write the approximate log unnormalized posterior as:

$$\log \tilde{q}(\mathbf{v}, \mathbf{x}_{0:T-1}) = \log |\mathbf{K}_Z|^{\frac{1}{2}} - \log |\mathbf{K}_Z|^{\frac{1}{2}} - \frac{1}{2}\mathbf{v}^\top \mathbf{v} + \log p(\mathbf{x}_0) + \log p(\mathbf{y}_0 \,|\, \mathbf{x}_0) + \sum_{t=1}^{T-1} \left[\log p(\mathbf{y}_t \,|\, \mathbf{x}_t) + \log \mathcal{N}(\mathbf{x}_t; \mathbf{K}_{t-1,Z}(\mathbf{L}_Z^{-1})^T \mathbf{v}, \mathbf{Q}) - \frac{1}{2}\operatorname{Tr}(\mathbf{Q}^{-1}\mathbf{B}_{t-1}) \right].$$

$$= \log p(\mathbf{v}) + \log p(\mathbf{x}_0) + \log p(\mathbf{y}_0 \,|\, \mathbf{x}_0) + \sum_{t=1}^{T-1} \left[\log p(\mathbf{y}_t \,|\, \mathbf{x}_t) + \log \mathcal{N}(\mathbf{x}_t; \mathbf{K}_{t-1,Z}(\mathbf{L}_Z^{-1})^T \mathbf{v}, \mathbf{Q}) - \frac{1}{2}\operatorname{Tr}(\mathbf{Q}^{-1}\mathbf{B}_{t-1}) \right]. \quad (66)$$

F. Collapsed Method

We can integrate out $\mathbf{u}$ in Eq. (21) to obtain the marginal distribution of $\mathbf{x}_{1:T}$.

$$q^*(\mathbf{x}_{1:T}) = \int_{\mathbf{u}} q^*(\mathbf{u}, \mathbf{x}_{1:T}) \mathrm{d}\mathbf{u} = \int_{\mathbf{u}} p(\mathbf{u} \,|\, \mathbf{Z}) p(\mathbf{x}_0) \prod_{t=1}^{T} \left[p(\mathbf{y}_t \,|\, \mathbf{x}_t) \exp(-\frac{1}{2}\operatorname{Tr}(\mathbf{Q}^{-1}\mathbf{B}_{t-1})) \mathcal{N}(\mathbf{x}_t; \boldsymbol{\mu}_{x_t}, \mathbf{Q}) \right] \mathrm{d}\mathbf{u} \quad (67)$$

$$= p(\mathbf{x}_0) \prod_{t=1}^{T} \left[p(\mathbf{y}_t \,|\, \mathbf{x}_t) \exp(-\frac{1}{2}\operatorname{Tr}(\mathbf{Q}^{-1}\mathbf{B}_{t-1})) \right] \int_{\mathbf{u}} p(\mathbf{u}) \prod_{t=1}^{T} \mathcal{N}(\mathbf{x}_t; \boldsymbol{\mu}_{x_t}, \mathbf{Q}) \mathrm{d}\mathbf{u}. \quad (68)$$

In the case of using our prior re-parameterization, i.e., whitened inducing variables, $\mathbf{v}$ and for multidimensional states and, therefore, multi-output GPs, we can write the integral above for dimension d as:

$$\int_{\mathbf{v}^{(d)}} p(\mathbf{v}^{(d)}) \prod_{t=1}^{T} \mathcal{N}(\mathbf{x}_t^{(d)}; \mathbf{m}_{t-1}^{(d)} + \tilde{\mathbf{A}}_{t-1}^{(d)} \mathbf{v}^{(d)}, \mathbf{Q}_d) \mathrm{d}\mathbf{v}^{(d)}$$

$$= \prod_{t=1}^{T} \mathcal{N}(\mathbf{x}_t^{(d)}; \mathbf{m}_{t-1}^{(d)}, \mathbf{Q}_d) \exp\left[\frac{1}{2} (\mathbf{g}^{(d)})^\top (\mathbf{H}^{(d)})^{-1} \mathbf{g}^{(d)} \right] \left(\det(\mathbf{H}^{(d)}) \right)^{\frac{1}{2}} \quad (69)$$

where

$$\mathbf{g}^{(d)} := \left(\mathbf{I} + \sum_{t=1}^{T} (\tilde{\mathbf{A}}_{t-1}^{(d)})^\top \mathbf{Q}_d^{-1} (\tilde{\mathbf{A}}_{t-1}^{(d)}) \right)^{-1} \left(\sum_t (\mathbf{x}_t^{(d)} - \mathbf{m}_{t-1}^{(d)})^\top \mathbf{Q}_d^{-1} \tilde{\mathbf{A}}_{t-1}^{(d)} \right)^\top, \quad (70)$$

$$\mathbf{H}^{(d)} = \left(\mathbf{I} + \sum_{t=1}^{T} (\tilde{\mathbf{A}}_{t-1}^{(d)})^\top \mathbf{Q}_d^{-1} (\tilde{\mathbf{A}}_{t-1}^{(d)}) \right)^{-1}$$

$$\tilde{\mathbf{A}}_{t-1}^{(d)} = \mathbf{K}_{t-1,Z}^{(d)} ((\mathbf{L}_Z^{(d)})^\top)^{-1}, \quad (71)$$

$$\mathbf{K}_Z^{(d)} = \mathbf{L}_Z^{(d)} (\mathbf{L}_Z^{(d)})^\top. \quad (72)$$

Thus, our optimal log variational posterior marginal over state trajectories $\mathbf{x}_{0:T}$ is

$$\log q^*(\mathbf{x}_{0:T}) = \log p(\mathbf{x}_0) + \sum_{t=1}^{T} \left[\log p(\mathbf{y}_t \,|\, \mathbf{x}_t) + \log \mathcal{N}(\mathbf{x}_t; \mathbf{m}_{t-1}, \mathbf{Q}) \right]$$

$$- \frac{1}{2} \sum_{d=1}^{d_x} \left[\sum_{t=1}^{T} \operatorname{Tr}(\mathbf{Q}_d^{-1} \mathbf{B}_{t-1}^{(d)}) + \log \det(\mathbf{H}^{(d)})^{-1} - (\tilde{\mathbf{x}}^{(d)})^\top \mathbf{H}^{(d)} (\tilde{\mathbf{x}}^{(d)}) \right], \quad (73)$$

where $\tilde{\mathbf{x}}^{(d)} := \sum_{t=1}^{T} (\tilde{\mathbf{A}}_{t-1}^{(d)})^\top \mathbf{Q}_d^{-1} (\mathbf{x}_t^{(d)} - \mathbf{m}_{t-1}^{(d)})$.

Algorithm 1 Particle MCMC for inferring latent states $\mathbf{x}_{0:T}$

Require: Number of particles S, observations $\mathbf{y}_{1:T}$, samples of $\mathbf{x}'_{1:T}$ in the previous iteration, likelihood parameters $\boldsymbol{\phi} = \{\mathbf{C}, \mathbf{d}, \mathbf{R}\}$,
  Initialize state $\mathbf{x}_0$, weights $W_t^{(i)} = 1$ for $i = 1, \ldots, S$.
  Fix the last particle as setting $\mathbf{x}_{1:T}^{(S)} = \mathbf{x}'_{1:T}$
  for $t = 1, \ldots, T$ **do**
    Generate $\mathbf{x}_t^{(i)}$ from Eq. (22) for $i = 1, \ldots, S-1$.
    Calculate weight $W_t^i = p_{\boldsymbol{\phi}}(\mathbf{y}_t | \mathbf{x}_t^{(i)})$ for $i = 1, \ldots, S$
    Normalize the weights as $\overline{W}_t^{(i)} = W_t^{(i)} / \sum_i W_t^{(i)}$
    if $t < T$ **then**
      For $i' = 1, \ldots, S-1$, re-sample the index $j_{i'}$ from the categorical distribution, with the event probabilities being $(\overline{W}_t^{(1)}, \ldots, \overline{W}_t^{(S)})$.
      For $i' = 1, \ldots, S-1$, let $\mathbf{x}_{1:t}^{(i')} := \mathbf{x}_{1:t}^{(j_{i'})}$ and set $W_t^{i'} = 1/S$
    else
      Re-sample the index j^* from the categorical distribution, with the event probabilities being $(\overline{W}_t^{(1)}, \ldots, \overline{W}_t^{(S)})$.
      Return $\mathbf{x}_{0:T} :== \mathbf{x}_{1:T}^{(j^*)}$
    end if
  end for

G. Closed-Form Optimal Variational Conditional Distribution over Inducing Variables

Given a trajectory, the optimal variational conditional over the whitened inducing variables given a state trajectory is given by:

$$q(\mathbf{v} \,|\, \mathbf{x}_{0:T}) = \prod_{d=1}^{d_x} \mathcal{N}(\mathbf{v}^{(d)}; \mathbf{g}^{(d)}, \mathbf{H}^{(d)}), \tag{74}$$

where $\mathbf{g}$ and $\mathbf{H}$ are given in Eqs. (70) and (71), respectively. A similar expression can be obtained for the original (non-whitened) inducing variables $\mathbf{u}$. Of importance here is that our formulation has allowed us to obtain an optimal conditional variational distribution, given the state trajectories, over the inducing variables in closed form. We see then that, in effect, our approximate posterior is $q_{\text{FFVD}}(\mathbf{x}_{0:T}, \mathbf{u}) = q(\mathbf{x}_{0:T}) q(\mathbf{u} \,|\, \mathbf{x}_{0:T})$. In order to make predictions, we run SGHMC to obtain samples $\{\mathbf{x}_{0:T}^{(s)}\}$ from the marginal $p(\mathbf{x}_{0:T})$ and then use the closed-form expression for the conditional to obtain $\{\mathbf{u}^{(s)} \,|\, \mathbf{x}_{0:T}^{(s)}\}$. Running SGHMC over the much lower-dimensional space of state trajectories (instead of the joint space of trajectories and inducing variables) should generally converge faster. Or experiments with both versions of the algorithm confirm this.

G.1. Optimal Closed-Form Conditional vs Assumed Parametric Factorizations

We contrast our variational distribution with that of the variational Gaussian process state space model (VGPSSM; Frigola et al., 2014) and the variationally coupled dynamic trajectories (VCDT; Ialongo et al., 2019). While, implicitly, our variational distribution is $q_{\text{FFVD}}(\mathbf{x}_{0:T}, \mathbf{u}) := q(\mathbf{x}_{0:T}) q(\mathbf{u} \,|\, \mathbf{x}_{0:T})$ the VGPSSM and VCDT assume $q_{\text{VGPSSM}}(\mathbf{x}_{0:T}, \mathbf{u}) := q(\mathbf{x}_{0:T}) q(\mathbf{u})$ and $q_{\text{VCDT}}(\mathbf{x}_{0:T}, \mathbf{u}) := q(\mathbf{x}_{0:T} \,|\, \mathbf{u}) q(\mathbf{u})$, respectively. Although the former (VGPSSM) obtained an "optimal" posterior (given the factorization assumption), it is a mean-field approximation and ignores the dependencies between state trajectories and inducing variables. The latter (VCDT), does not make a mean-field assumption but its factorization forces parametric constraints over the individual distributions. In fact, Ialongo et al. (2019) assume Gaussian posteriors for both the marginal $q(\mathbf{u})$ and each of the time-dependent conditionals in $q(\mathbf{x}_{0:T} \,|\, \mathbf{u})$. Our proposal is the only variational distribution that is theoretically optimal while yielding a closed-form conditional.

H. Particle Markov chain Monte Carlo (MCMC)

We can use the sequential structure of the latent states $\mathbf{x}_{0:T}$ for efficient inference. More specifically, using the transition in Eq. (22). The details of this are given in Algorithm 1.

I. Experiment Details

Regarding the variational Gaussian process state space model (VGPSSM), probabilistic recurrent state space model (PRSSM), and VCDT, we download the authors' implementations from their websites. For a fair comparison, we try to follow the same treatment for hyper-parameters across all methods. We note, however, that the Matlab implementation of VGPSSM does not provide inference for GP hyper-parameters $\boldsymbol{\theta}$, conditional likelihood parameters $\boldsymbol{\phi}$, inducing inputs $\mathbf{Z}$, process variance Q or observation variance $\mathbf{R}$.

I.1. Settings for synthetic data

In generating the synthetic data, we set the kernel's signal variance $\sigma = 2.0$ and lengthscale $l = 0.5$ in the sparse GPSSM. we reduce the impact of observational error and set the observation variance as small as $\sigma^2 = 0.01$, i.e., each observation would be generated as $y_t \sim \mathcal{N}(x_t, 0.01)$. We also set the process variance $\mathbf{Q} = 0.01$. The number of inducing points is set to 20, and these 20 inducing points are evenly spread in the interval $[-2, 2]$. We set the length of the training trajectory as 120, the number of iterations as 50 000, and the number of posterior samples as 50.

I.1.1. Testing the Marginals for Gaussianity

We conduct a hypothesis test for each individual trajectory state and inducing variable. Given the 50 posterior samples for $\mathbf{x}_t$ (or $\mathbf{u}_m$), which we denote them as $\{\mathbf{x}_t^{(s)}\}_{s=1}^{50}$ (or $\{\mathbf{u}_m^{(s)}\}_{s=1}^{50}$), we use the implementation of *scipy.stats.normaltest* from Python's *scipy* package to test whether we have sufficient evidence to reject the hypothesis that $\mathbf{x}_t$ (or $\mathbf{u}_m$) follows a Gaussian distribution. *scipy.stats.normaltest* is based on the work of D'agostino & Pearson (1973).

I.2. Settings for Real-world Data

We adopt similar settings as in Ialongo et al. (2019) to initialize the hyper-parameters for all the models, which first uses a factorized nonlinear model to optimize $\boldsymbol{\theta}, \mathbf{x}_{0:T}, \mathbf{u}, \mathbf{Z}, \mathbf{C}, \mathbf{d}, \mathbf{R}, \mathbf{Q}$. We use the identity function as the mean function and the squared exponential function with automatic relevance determination (ARD) in GPSSM. When dealing with more than 1 latent dimension, which requires a multi-output GP, we use a different set of kernel hyper-parameters for each output. We set the number of inducing points to $M = 100$ and the number of dimensions for latent states $\mathbf{x}_{0:T}$ to $d_x = 4$. We set standard diagonal Gaussian priors for the initial latent state $\mathbf{x}_0$ and (where applicable) for likelihood parameters $\mathbf{C}, \mathbf{d}$, the logarithm of observation standard deviation $\log(\mathbf{R})/2$ and process variance $\log(\mathbf{Q})$.

For optimizing these hyper-parameters, we use Adam, with default settings for the optimizer parameters except for a decayed learning rate. We run our FFVD algorithm for 50 000 iterations and VCDT for 200 000. We set the learning rate as 0.01 and the decay parameter as 0.05 during SGHMC sampling. We used $S = 10^5$ posterior samples for VCDT (as in the results in the original paper) and $S = 100$ for ours. Following the evaluation in Ialongo et al. (2019), we use the RMSE between the models' predictions and the ground truth in the future 30 time steps (not seen in training) as the comparison criterion for all the methods.

Regarding the benchmarks, we have 1 024 observations for *Actuator*, 1 000 observations for *Ballbeam*, 500 observations for *Drive*, 1 000 observations for *Dryer*, 1 024 for *Flutter*, and 296 observations for *Furnace*. Training lengths are: $T_{\text{train}} = 500, 500, 250, 500, 500, 150$, respectively and test lengths are the rest of the sequences. All these observations are 1-dimensional but, as in previous work, we consider $d_x = 4$-dimensional latent states. Each benchmark also contains a 1-dimensional control input vector $\mathbf{a} \in \mathbb{R}^{T \times 1}$.

I.3. Details of Performance Metrics

We use the test RMSE and the negative mean log likelihood (NMLL).

Given samples from the predictive distribution $\{\hat{\mathbf{y}}_t^{(s)}\}$ we can compute the RMSE as:

$$\text{RMSE} = \frac{1}{T} \sum_{t=1}^{T} \left(\mathbf{y}_t - \mathbb{E}\{\hat{\mathbf{y}}_t^{(s)}\} \right)^2, \tag{75}$$

where $\mathbb{E}\{\hat{\mathbf{y}}_t^{(s)}\}$ is estimated as the empirical mean.

if we the corresponding method gives us $\{\mathbf{x}_t^{(s)}\}$, then we can estimate the NMLL as:

$$\text{NMLL} = -\frac{1}{S}\frac{1}{T}\sum_{s=1}^{S}\sum_{t=1}^{T}\log\mathcal{N}(\mathbf{y}_t; \mathbf{C}\mathbf{x}_t^{(s)} + \mathbf{d}, \mathbf{R}). \tag{76}$$

If the method only provides us with $\{\hat{\mathbf{y}}_t^{(s)}\}$ then a reasonable estimate is:

$$\text{NMLL} = -\frac{1}{T}\sum_{t=1}^{T}\log\mathcal{N}(\mathbf{y}_t; \mathbb{E}\{\hat{\mathbf{y}}_t^{(s)}\}, \mathbb{V}\{\hat{\mathbf{y}}_t^{(s)}\}), \tag{77}$$

where $\mathbb{E}\{\hat{\mathbf{y}}_t^{(s)}\}, \mathbb{V}\{\hat{\mathbf{y}}_t^{(s)}\}$ are the empirical mean and variance of the predicted samples.

For FFVD, we simply use:

$$\text{NMLL}_{\text{FFVD}} = -\frac{1}{S}\frac{1}{T}\sum_{s=1}^{S}\sum_{t=1}^{T}\log\mathcal{N}(\mathbf{y}_t; \mathbf{C}\boldsymbol{\mu}_t^{(s)} + \mathbf{d}, \mathbf{C}\boldsymbol{\Sigma}_t^{(s)}\mathbf{C}^\top + \mathbf{R}), \tag{78}$$

where $\boldsymbol{\mu}_t^{(s)}$ and $\boldsymbol{\Sigma}_t^{(s)}$ are given by Eq. (22).

I.3.1. Data Normalization

The normalization (i.e., standardization) of observations is given by the transformation:

$$\tilde{y}_t = \frac{y_t - m_y}{\sigma_y}, \tag{79}$$

where m_y and σ_y are the mean and standard deviation of the training data. Let us denote the $\widetilde{\text{RMSE}}$ and $\widetilde{\text{NMLL}}$ as the error metrics in the normalized space. It is easy to show we can obtain the metrics in the original space as:

$$\text{RMSE} = \sigma_y\widetilde{\text{RMSE}} \tag{80}$$

$$\text{NMLL} = \log\sigma_y + \widetilde{\text{NMLL}}. \tag{81}$$

I.4. Reproducibility

We will make our code publicly available upon acceptance of the paper.

J. Additional Results

J.1. Training and Test Performance on All Benchmarks

Fig. 5 shows the training and testing regimes and predictions for the six system identification benchmarks considered.

J.2. Traceplot of Log-Likelihood on Training data for FFVD-M and FFVD-C-M

Figure 6 shows the convergence trace plots for all benchmarks.

K. Extended Related Work

Here we give more details about the differences between our method and closely related approaches for inference in the GPSSM. The VGPSSM of Frigola et al. (2014) is the first developing variational approaches to inference in GPSSMs, and it uses mean-field assumptions to factorize the variational distribution of inducing variables $\mathbf{u}$ and latent state trajectories $\mathbf{x}_{0:T}$ as $q_{\text{VGPSSM}}(\mathbf{u}, \mathbf{x}_{0:T}) = q(\mathbf{u})q(\mathbf{x}_{0:T})$. This assumption clearly overlooks the complex dependencies between $\mathbf{u}$ and $\mathbf{x}_{0:T}$. Doerr et al. (2018) model these dependencies with their PRSSM algorithm, but make the unrealistic assumption of the posterior dynamics over the latent state trajectories being the same as the prior. Ialongo et al. (2019) show that this assumption can have critical consequences on the posterior and predictive distributions and propose a more general

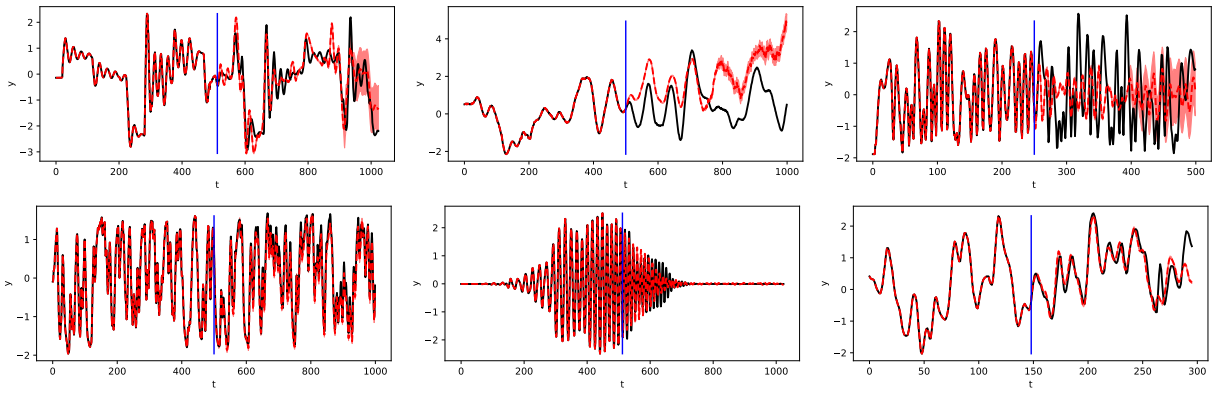

Figure 5. Training and test performance on benchmarks. Row 1 (left to right): *Actuator*, *Ballbeam*, *Drive*; row 2 (left to right): *Dryer*, *Flutter*, *Furnace*.

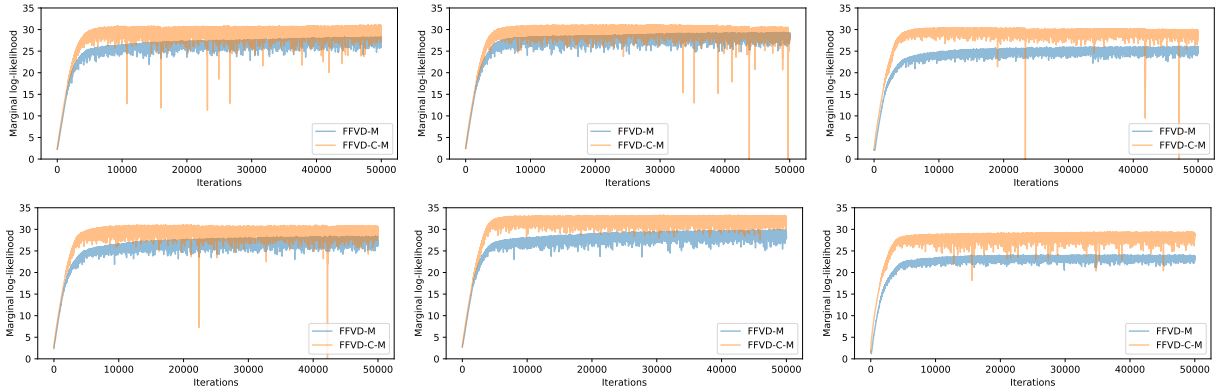

Figure 6. Traceplot of log-likelihood on training data for FFVD-M and FFVD-C-M. Row 1 (left to right): *Actuator*, *Ballbeam*, *Drive*; row 2 (left to right): *Dryer*, *Flutter*, *Furnace*.

algorithm called VCDT. Although a significant improvement over previous methods, VCDT assumes posterior Gaussian distributions. When considering our main object of interest, i.e., the posterior over the state trajectories, under the standard and most commonly used GP setting with non-linear kernels, a Gaussian posterior assumption never holds, as the latent states are inputs to the kernel. Hence, our approach is theoretically superior to VCDT and provides a new baseline with broad practical applicability.

As mentioned in the main paper, Frigola et al. (2013) also proposed a PMCMC algorithm for inference over state trajectories. However, the algorithm is based on the fully-independent conditional approximation (FIC; Quinonero-Candela & Rasmussen, 2005). As reported in Frigola et al. (2013), a naïve implementation of their algorithm has a time complexity of $\mathcal{O}(M^2T^2)$, which is significantly higher than ours. A much more efficient implementation with $\mathcal{O}(M^2T)$ time complexity that requires significant tracking of intermediate data structures and factorizations is hinted at in the original paper. Unfortunately, the corresponding code has not been made publicly available. Furthermore, model approximations such as those in FIC can have surprising consequences, such as incorrectly estimating the noise variance to be almost zero or ignoring additional inducing inputs, see, e.g., Bauer et al. (2016) for a thorough discussion. Consequently, we see our variational approach as more principled.