# A General Method for Amortizing Variational Filtering


Joseph Marino, Milan Cvitkovic, Yisong Yue

*California Institute of Technology (Caltech)*

Caltech


## Problem

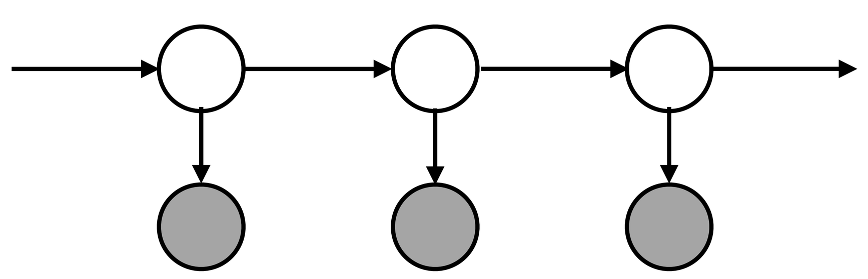
Deep latent variable models are often used for dynamical tasks, like reinforcement learning or time-series prediction. A central challenge is performing efficient online inference of the hidden states (filtering). In the static setting, amortized variational techniques are widely used for inference, but applying these techniques to dynamical problems has required hand-crafting an inference procedure for every new model.
**We propose a general purpose method for *efficiently* performing *accurate* inference in *any* dynamical latent variable model.**

## Background

A dynamical latent variable model models a sequence of observations, $\mathbf{x}_{\leq T}$, using a sequence of latent variables, $\mathbf{z}_{\leq T}$, and parameters, $\theta$. These models are of the general form:

$$p_\theta(\mathbf{x}_{\leq T}, \mathbf{z}_{\leq T}) = \prod_{t=1}^{T} p_\theta(\mathbf{x}_t | \mathbf{x}_{<t}, \mathbf{z}_{\leq t}) p_\theta(\mathbf{z}_t | \mathbf{x}_{<t}, \mathbf{z}_{<t}).$$

$p_\theta(\mathbf{x}_t | \mathbf{x}_{<t}, \mathbf{z}_{\leq t})$ is the *observation model*, and $p_\theta(\mathbf{z}_t | \mathbf{x}_{<t}, \mathbf{z}_{<t})$ is the *dynamics model*. A simplified version of such models can be represented graphically as:



## Variational Filtering

Given a sequence of observations, we want to infer the posterior distribution over the sequence of latent variables, $p_\theta(\mathbf{z}_{\leq T} | \mathbf{x}_{\leq T})$. This is often intractable. Instead, we use an approximate posterior, $q(\mathbf{z}_{\leq T} | \mathbf{x}_{\leq T})$, and minimize the following variational objective, called the ***free energy***:

$$\mathcal{F} \equiv -\mathbb{E}_{q(\mathbf{z}_{\leq T} | \mathbf{x}_{\leq T})} \left[ \log \frac{p_\theta(\mathbf{x}_{\leq T}, \mathbf{z}_{\leq T})}{q(\mathbf{z}_{\leq T} | \mathbf{x}_{\leq T})} \right].$$

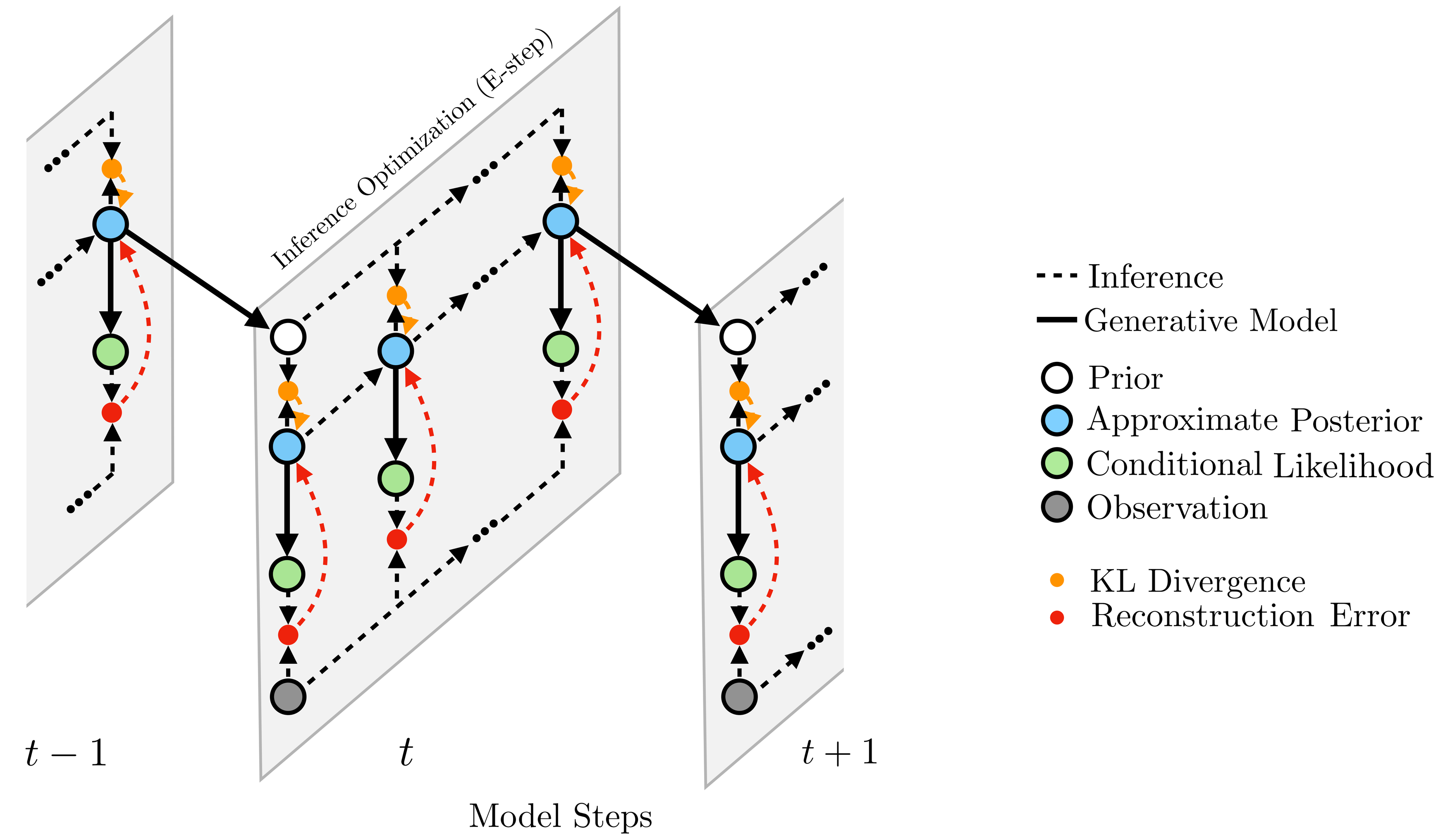
We assume the ***filtering*** setting, where only past and present variables are used for inference, and assume the approximate posterior factorizes as

$$q(\mathbf{z}_{\leq T} | \mathbf{x}_{\leq T}) = \prod_{t=1}^{T} q(\mathbf{z}_t | \mathbf{x}_{\leq t}, \mathbf{z}_{<t}).$$

With this filtering approximate posterior, the free energy becomes:

$$\mathcal{F} = \sum_{t=1}^{T} \mathbb{E}_{\prod_{\tau=1}^{t-1} q(\mathbf{z}_\tau | \mathbf{x}_{\leq \tau}, \mathbf{z}_{<\tau})} [\mathcal{F}_t]$$

$$\mathcal{F}_t \equiv -\mathbb{E}_{q(\mathbf{z}_t | \mathbf{x}_{\leq t}, \mathbf{z}_{<t})} \left[ \log \frac{p_\theta(\mathbf{x}_t, \mathbf{z}_t | \mathbf{x}_{<t}, \mathbf{z}_{<t})}{q(\mathbf{z}_t | \mathbf{x}_{\leq t}, \mathbf{z}_{<t})} \right]$$



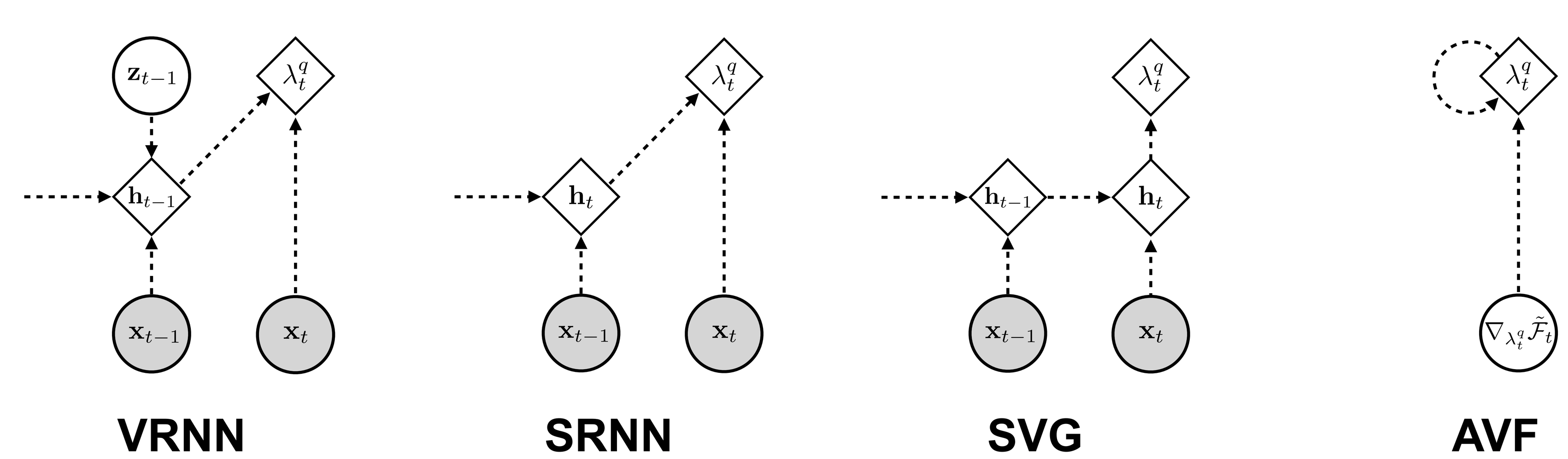

**Algorithm 1** Variational Filtering Expectation Maximization

**Input:** observation sequence $\mathbf{x}_{1:T}$, model $p_\theta(\mathbf{x}_{1:T}, \mathbf{z}_{1:T})$
$\nabla_\theta \mathcal{F} = 0$
**for** $t = 1$ **to** $T$ **do**
  initialize $q(\mathbf{z}_t | \mathbf{x}_{\leq t}, \mathbf{z}_{<t})$ ▷ from $p_\theta(\mathbf{z}_t | \mathbf{x}_{<t}, \mathbf{z}_{<t})$
  $\tilde{\mathcal{F}}_t := \mathbb{E}_{q(\mathbf{z}_{<t} | \mathbf{x}_{<t}, \mathbf{z}_{<t-1})} [\mathcal{F}_t]$
  $q(\mathbf{z}_t | \mathbf{x}_{\leq t}, \mathbf{z}_{<t}) = \arg\min_q \tilde{\mathcal{F}}_t$ ▷ inference (E-step)
  $\nabla_\theta \mathcal{F} = \nabla_\theta \mathcal{F} + \nabla_\theta \tilde{\mathcal{F}}_t$
**end for**
$\theta = \theta - \alpha \nabla_\theta \mathcal{F}$ ▷ learning (M-step)

The ***variational filtering EM*** algorithm minimizes the filtering free energy by sequentially minimizing the free energy at each step. Initializing the approximate posterior at each step from the prior yields a Bayesian ***prediction-update*** loop.

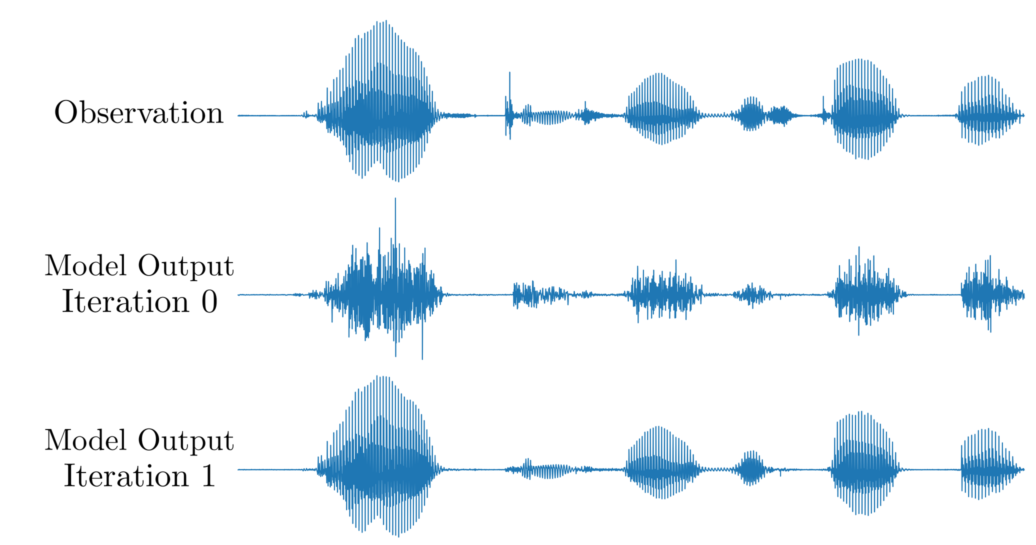
We can amortize inference optimization by using an iterative inference model at each step (Marino *et al.*, 2018), which we refer to as ***amortized variational filtering***. With the approximate posterior parameters at step $t$ as $\boldsymbol{\lambda}_t^q$, the inference update is

$$\boldsymbol{\lambda}_t^q \leftarrow f_\phi(\boldsymbol{\lambda}_t^q, \nabla_{\boldsymbol{\lambda}_t^q} \tilde{\mathcal{F}}_t).$$

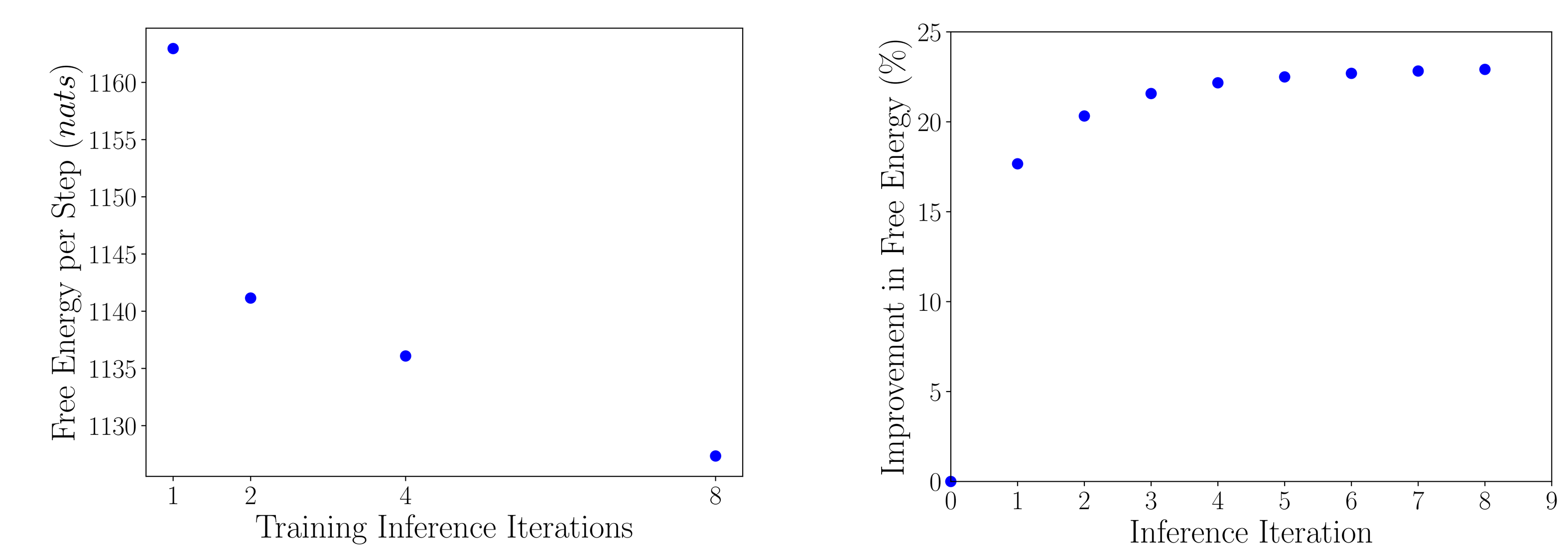



## Results

We evaluate amortized variational filtering (**AVF**) on three dynamical latent variable models: **VRNN** (Chung *et al.*, 2015), **SRNN** (Fraccaro *et al.*, 2016), and **SVG** (Denton & Fergus, 2018). We train these models on speech, MIDI music, and video data sets.




**Qualitative Evaluation** - We demonstrate the effect of additional inference iterations with AVF on the TIMIT data set. **Above**: Output predictions and reconstructions. **Below**: Validation performance and inference improvement vs. inference iterations.




**Quantitative Evaluation** - We compare AVF with baseline filtering methods for VRNN, SRNN, and SVG. We find that, in all cases, AVF results in improved model performance in terms of average free energy.

| **Speech** | TIMIT |
|---|---|
| VRNN | |
| baseline | 1,082 |
| AVF (1 step) | 1,105 |
| AVF (2 step) | **1,071** |
| SRNN | |
| baseline | 1,026 |
| AVF (1 step) | **1,024** |

| **Video** | KTH Actions |
|---|---|
| SVG | |
| baseline | 15,097 |
| AVF (1 step) | **11, 714** |

| **Music** | Piano-midi.de | MuseData | JSB Chorales | Nottingham |
|---|---|---|---|---|
| SRNN | | | | |
| baseline [Fraccaro *et al.*] | 8.20 | 6.28 | 4.74 | 2.94 |
| baseline | 8.19 | 6.27 | 6.92 | 3.19 |
| AVF (1 step) | **8.12** | **5.99** | 6.97 | **3.13** |
| AVF (5 step) | – | – | **6.77** | – |

**Code**: `github.com/joelouismarino/amortized-variational-filtering`