Entropic Matching for Expectation Propagation of Markov Jump Processes

Yannick Eich **Bastian Alt** **Heinz Koeppl**
Department of Electrical Engineering and Information Technology
Technische Universität Darmstadt
{yannick.eich, heinz.koeppl}@tu-darmstadt.de

Abstract

We propose a novel, tractable latent state inference scheme for Markov jump processes, for which exact inference is often intractable. Our approach is based on an entropic matching framework that can be embedded into the well-known expectation propagation algorithm. We demonstrate the effectiveness of our method by providing closed-form results for a simple family of approximate distributions and apply it to the general class of chemical reaction networks, which are a crucial tool for modeling in systems biology. Moreover, we derive closed-form expressions for point estimation of the underlying parameters using an approximate expectation maximization procedure. We evaluate our method across various chemical reaction networks and compare it to multiple baseline approaches, demonstrating superior performance in approximating the mean of the posterior process. Finally, we discuss the limitations of our method and potential avenues for future improvement, highlighting its promising direction for addressing complex continuous-time Bayesian inference problems.

1 INTRODUCTION

Markov jump processes (MJPs) play a crucial role for modeling diverse phenomena in various domains, including finance (Mamon and Elliott, 2007), engineering (Bolch et al., 2006), and biology (Anderson and Kurtz, 2015). In the field of systems biology, MJPs find particular significance, offering powerful modeling capabilities for complex systems, such as chemical reaction networks (CRNs) (Anderson and Kurtz, 2015). By incorporating prior knowledge of the dynamic nature of underlying processes, MJP models can efficiently extract valuable insights and facilitate understanding and control of these intricate systems. This becomes particularly crucial when dealing with latent processes, where only partial information about the desired quantities of interest is available. The resulting inverse problem presents significant challenges, requiring the solution of the underlying Bayesian filtering and smoothing problem.

Traditional approaches to latent state inference in continuous-time stochastic processes often rely on system approximations using ordinary differential equations (ODEs) or stochastic differential equations (SDEs), see, e.g., (Gardiner et al., 1985). However, inference methods based on Kalman filtering and RTS smoothing that exploit linearization procedures, such as the linear noise approximation (Komorowski et al., 2009), can suffer from the limitations of the underlying non-linear rate function. Similarly, approaches based on the non-linear chemical Langevin equation (Gillespie, 2000) may yield inaccurate results, especially in scenarios characterized by low counting numbers.

In contrast, modeling the process directly using an MJP model captures discrete state transitions and provides a more accurate representation of the underlying dynamics. However, latent state inference in MJPs often requires computationally demanding sampling-based techniques, such as sequential Monte Carlo (SMC) (Doucet et al., 2001; Golightly and Wilkinson, 2011), which suffers from the well known particle degeneracy problem, especially for long trajectories. More recently, particle Markov chain Monte Carlo (MCMC) methods (Andrieu et al., 2010; Golightly et al., 2015; Lowe et al., 2023) have shown promising results in Bayesian parameter estimation; however, the underlying latent state estimation still relies on SMC, inheriting its limitations.

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258. Copyright 2025 by the author(s).

Alternatively, deterministic methods based on variational inference (VI) provide valuable tools. Here, the posterior process is approximated using tractable approximations, such as mean-field VI (Opper and Sanguinetti, 2007) or moment-based VI (Wildner and Koeppl, 2019). More recently, Seifner and Sánchez (2023) proposed a neural variational inference method to jointly learn the parameters and the latent state. However, their latent state estimation relies on the integration of the chemical master equation, which is not scalable to large models.

We take a different path by employing a message passing method in continuous-time. Instead of approximating the distribution on a path-wise level, as seen in variational Bayesian methods, our approach involves approximating the exact message passing scheme and thereby optimizing the posterior marginal distributions. This allows us to accommodate for the MJP dynamics and to embed our method into the expectation propagation (EP) algorithm (Minka, 2001). An implementation of our proposed method is publicly available[1].

2 BACKGROUND

Markov Jump Processes. A Markov jump process (MJP) (Ethier and Kurtz, 2009) is a continuous-time Markov process $\{X(t) \in \mathcal{X} \mid t \in \mathbb{R}_{\geq 0}\}$ on a discrete state space $\mathcal{X}$. The Markov property implies that for all $t' > t$ we have $\mathrm{P}(X(t') = x(t') \mid \{X(s) = x(s) \mid s \in [0, t]\}) = \mathrm{P}(X(t') = x(t') \mid X(t) = x(t))$. Hence, an MJP is fully described by an initial distribution $p_0(x) = \mathrm{P}(X(0) = x)$, $\forall x \in \mathcal{X}$, and its rate function

$$\Lambda(x, x', t) \coloneqq \lim_{h \to 0} \frac{\mathrm{P}(X(t+h) = x' \mid X(t) = x)}{h},$$

for all time points $t \in \mathbb{R}_{\geq 0}$ and states $x \in \mathcal{X} \neq x' \in \mathcal{X}$. Given these quantities, one can derive a system of ODEs of the time-marginal probability function $p_t(x) \coloneqq \mathrm{P}(X(t) = x)$ of the MJP, which is given by the differential forward Chapman-Kolmogorov equation for $p_t(x)$, the *master equation*,

$$\frac{\mathrm{d}}{\mathrm{d}t} p_t(x) = [\mathcal{L}_t p_t](x), \quad \forall x \in \mathcal{X}, \tag{1}$$

with initial distribution $p_0(x)$ at time point $t = 0$, where the operator $\mathcal{L}_t$ is given as

$$[\mathcal{L}_t \phi](x) \coloneqq \sum_{x' \neq x} \Lambda(x', x, t)\phi(x') - \Lambda(x, x', t)\phi(x),$$

for an arbitrary test function ϕ. Additionally, an evolution equation for some arbitrary moment functions can be found, by multiplying Eq. (1) from the left with a moment function $s(x)$ and summing over all elements $x \in \mathcal{X}$, this yields ${}^{\mathrm{d}}/_{\mathrm{d}t}\, \mathsf{E}[s(X(t))] = \mathsf{E}[\mathcal{L}_t^\dagger s(X(t))]$, where $\mathcal{L}_t^\dagger$ is the adjoint operator of $\mathcal{L}_t$, w.r.t. the inner product $\langle \phi, \psi \rangle = \sum_{x \in \mathcal{X}} \phi(x)\psi(x)$ given as

$$[\mathcal{L}_t^\dagger \psi](x) \coloneqq \sum_{x' \neq x} \Lambda(x, x', t)[\psi(x') - \psi(x)],$$

for an arbitrary test function ψ. For more on MJPs see (Ethier and Kurtz, 2009; Gardiner et al., 1985; Norris, 1998; Del Moral and Penev, 2017).

Chemical Reaction Networks. We focus on chemical reaction networks (CRNs) (Anderson and Kurtz, 2015) with the species $\{\mathsf{X}_i \mid i = 1, \ldots, n\}$ and k reactions given as

$$\underline{\nu}_{1j}\mathsf{X}_1 + \cdots + \underline{\nu}_{nj}\mathsf{X}_n \xrightarrow{c_j} \bar{\nu}_{1j}\mathsf{X}_1 + \cdots + \bar{\nu}_{nj}\mathsf{X}_n,$$

$\forall j = 1, \ldots, k$, where $c_j \in \mathbb{R}_{\geq 0}$ is the reaction rate of the jth reaction. The substrate and product stoichiometries are given by the matrices $\underline{\nu} \in \mathbb{N}_0^{n \times k}$ and $\bar{\nu} \in \mathbb{N}_0^{n \times k}$, respectively. The state $X(t) \in \mathcal{X}$ of the network at time point t is described by the amount of each of the n species $X(t) = [X_1(t), \ldots, X_n(t)]^\top$, with $X_i(t) \in \mathcal{X}_i \subseteq \mathbb{N}_0$, $\forall i = 1, \ldots, n$, and $\mathcal{X} = \bigtimes_{i=1}^n \mathcal{X}_i \subseteq \mathbb{N}_0^n$. For CRNs one assumes that the stochastic process $\{X(t) \mid t \in \mathbb{R}_{\geq 0}\}$ is an MJP, with rate function

$$\Lambda(x, x', t) = \sum_{j=1}^k \mathbb{1}(x' = x + \nu_j)\lambda_j(x), \tag{2}$$

where the change vector $\nu_j \in \mathbb{Z}^n$ corresponding to the jth reaction is $\nu_j = \bar{\nu}_{\cdot j} - \underline{\nu}_{\cdot j}$. We assume that the propensity function $\lambda_j(x)$ corresponding to the jth reaction is given by mass action kinetics as

$$\lambda_j(x) = \bar{c}_j \prod_{i=1}^n \binom{x_i}{\underline{\nu}_{ij}} = c_j \prod_{i=1}^n (x_i)_{\underline{\nu}_{ij}}, \tag{3}$$

where the factors $\{\underline{\nu}_{1j}!, \ldots, \underline{\nu}_{nj}!\}$ are absorbed in the jth reaction rate coefficient as $c_j = {}^{\bar{c}_j}/_{\prod_{i=1}^n \underline{\nu}_{ij}!}$ and $(m)_n \coloneqq \frac{m!}{(m-n)!} = \prod_{k=0}^{m-1}(n-k)$ denotes the falling factorial. For more on CRNs see (Gardiner et al., 1985; Anderson and Kurtz, 2015; Wilkinson, 2018).

3 EXACT INFERENCE FOR LATENT MARKOV JUMP PROCESSES

We consider continuous-discrete inference (Maybeck, 1982; Huang et al., 2016; Särkkä and Solin, 2019) for latent MJPs. The model is given by the latent MJP $\{X(t) \in \mathcal{X} \mid t \in \mathbb{R}_{\geq 0}\}$ on $\mathcal{X}$ characterized via its rate function $\Lambda(x, x', t)$ and its initial probability distribution $p_0(x)$. The latent state $X(t)$ is not directly

[1] `https://github.com/yannickeich/ep4crns`

observed, rather we consider a discrete-time observation model with N observations $\{Y_1, \ldots, Y_N\}$ at time points $t_1 < t_2 < t_3 < \cdots < t_N$ as

$$Y_i \mid \{X(t_i) = x\} \sim p(y_i \mid x), \quad \forall i = 1, \ldots, N.$$

The problem of inferring the latent MJP $X(t)$ at time point t given some observations $y_{[0,T]} \coloneqq \{y_1, \ldots, y_N\}$ in a time interval $[0, T]$ is cast as a continuous-time Bayesian filtering and smoothing problem (Pardoux, 1981; Anderson and Rhodes, 1983; Särkkä and Solin, 2019), similar to the discrete-time setting (Särkkä, 2013). To this end, the two elementary objects are the filtering $\pi_t(x)$ and smoothing distribution $\tilde{\pi}_t(x)$, which are defined as

$$\begin{aligned} \pi_t(x) &\coloneqq \mathrm{P}(X(t) = x \mid y_{[0,t]}), \\ \tilde{\pi}_t(x) &\coloneqq \mathrm{P}(X(t) = x \mid y_{[0,T]}), \quad \forall x \in \mathcal{X}. \end{aligned}$$

For the filtering distribution $\pi_t(x)$ we condition on the set $y_{[0,t]} \coloneqq \{y_1, \ldots, y_K\}$ of $K = \sum_{i=1}^{N} \mathbb{1}(t_i \leq t)$ observations in the interval $[0, t]$ up until time point t and for the smoothing distribution $\tilde{\pi}_t(x)$ we consider all observations $y_{[0,T]}$ in the whole interval $[0, T]$ up until time point T. The filtering distribution can be computed recursively. First, it is easy to notice that the filtering distribution at the initial time point $t = 0$ is the initial distribution of the latent MJP, i.e., $\pi_0(x) = \mathrm{P}(X(0) = x) = p_0(x)$. Additionally, a standard result in continuous-discrete filtering is that the filtering distribution in between observations follows the latent prior dynamics and at the observation time points is subject to discrete-time updates (Huang et al., 2016; Särkkä and Solin, 2019). Hence, we have a system of ODEs with reset conditions as

$$\begin{aligned} &\pi_0(x) = p_0(x), \\ &\frac{\mathrm{d}}{\mathrm{d}t}\pi_t(x) = [\mathcal{L}_t \pi_t](x), \\ &\pi_{t_i}(x) = \frac{p(y_i \mid x)\pi_{t_i^-}(x)}{\sum_{x' \in \mathcal{X}} p(y_i \mid x')\pi_{t_i^-}(x')}, \end{aligned} \tag{4}$$

$\forall x \in \mathcal{X}$ and $i = 1, \ldots, N$, where throughout we denote by $\phi(t_i^-) \coloneqq \lim_{t \nearrow t_i} \phi(t)$ the limit from the left for a left-continuous function ϕ. Note, that the filtering distribution $\pi_t(x)$ is a càdlàg process in time t. There are multiple options to compute an evolution equation for the smoothing distribution. Often a backward-filtering and subsequent forward-smoothing approach is used for continuous-time systems, see, e.g., (Archambeau and Opper, 2011; Wildner and Koeppl, 2019; Mider et al., 2021). For completeness, we discuss this approach in Appendix 1. However, later on we exploit a different approach, which considers a forward-filtering backward-smoothing (FFBS) scheme. First, it is easy to notice that the smoothing distribution has an end point condition, as the smoothing distribution at time point $t = T$ is equal to the filtering distribution, i.e., $\tilde{\pi}_T(x) = \mathrm{P}(X(T) = x \mid y_{[0,T]}) = \pi_T(x)$. For the FFBS scheme it is shown in (Anderson and Rhodes, 1983) that the smoothing distribution follows an ODE with the filter end-point condition as

$$\tilde{\pi}_T(x) = \pi_T(x), \quad \frac{\mathrm{d}}{\mathrm{d}t}\tilde{\pi}_t(x) = -[\tilde{\mathcal{L}}_t \tilde{\pi}_t](x), \tag{5}$$

where the backward smoothing operator $\tilde{\mathcal{L}}_t$ and the corresponding backward smoothing rate function $\tilde{\Lambda}(x', x, t)$ is given as

$$\begin{aligned} &[\tilde{\mathcal{L}}_t \phi](x) \coloneqq \sum_{x' \neq x} \tilde{\Lambda}(x', x, t)\phi(x') - \tilde{\Lambda}(x, x', t)\phi(x), \\ &\tilde{\Lambda}(x, x', t) = \Lambda(x', x, t)\frac{\pi_t(x')}{\pi_t(x)}, \end{aligned}$$

where ϕ is an arbitrary test function and $\tilde{\Lambda}(x, x', t)$ is defined for states $x \neq x'$ with $x, x' \in \mathcal{X}$. The backward smoothing rate function $\tilde{\Lambda}(x', x, t)$ depends on the filtering distribution $\pi_t(x)$. The ratio of filtering distributions appearing in the backward smoothing rate can be seen as a correction factor, when computing the dynamics of a backward Markov process (Elliott, 1986; Van Handel, 2007), similar to the score correction appearing for continuous state space systems (Anderson, 1982). Therefore, by integrating the filtering distribution forward in time and computing the smoothing distribution backward in time, we can solve the filtering and smoothing problem. However, the substantial down-side of the exact filtering and smoothing equations is that they are intractable. This can easily be seen, by noticing that the distributions are defined on the state space $\mathcal{X}$. Hence, all sums go over $|\mathcal{X}|$ elements and we have to solve the $|\mathcal{X}|$-dimensional system of ODEs (4) forwards in time and backwards in time in Eq. (5). For example, for CRNs with a state space $\mathcal{X} = \mathbb{N}_0^n$ this is even infinite dimensional. Additionally, even when truncating the state space, the complexity still scales exponentially in the dimensionality n. Hence, we are required to use an approximate inference method.

4 APPROXIMATE INFERENCE FOR LATENT MARKOV JUMP PROCESSES

We use an approximate inference method to perform latent state inference for MJP models. To this end we approximate both the filtering distribution and the smoothing distribution in a FFBS scheme by using a parametric distribution $q(x \mid \theta)$ as $\pi_t(x) \approx q(x \mid \theta(t))$ and $\tilde{\pi}_t(x) \approx q(x \mid \tilde{\theta}(t))$, where $\theta(t) \in \Theta \subseteq \mathbb{R}^p$ and $\tilde{\theta}(t) \in \Theta$ are variational parameters for the filtering and

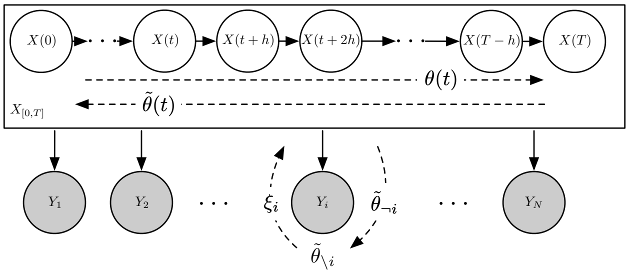

Figure 1: Probabilistic graphical model and approximate inference scheme for MJPs. The continuous-time Markov process $X_{[0,T]} = \{X(t) \mid t \in [0,T]\}$ emits the observations $\{Y_1, \dots, Y_N\}$. The approximate inference scheme consists of a continuous-time message passing algorithm, depicted by the dashed lines.

smoothing distribution, respectively. To find those variational parameters we use an assumed density method known as entropic matching (Ramalho et al., 2013; Bronstein and Koeppl, 2018). Therefore, we derive approximate filtering and smoothing equations for latent MJPs. Further, we exploit an EP algorithm (Minka, 2001) for inference and, therefore, extend the method of Cseke et al. (2016) from diffusion processes to MJPs. A probabilistic graphical model and the approximate inference scheme is depicted in Fig. 1

4.1 Entropic Matching for Filtering and Smoothing

We find the variational filtering parameters $\theta(t)$ by considering the minimization of the inclusive Kullback-Leibler (KL) divergence as $\theta(t+h) = \operatorname{argmin}_{\theta'} \mathsf{KL}(\pi_{t+h}(x) \parallel q(x \mid \theta'))$, where $\pi_{t+h}(x)$ is the propagated filtering distribution and h is a small time step. By considering that the previous filtering distribution at time point t is approximately $q(x \mid \theta(t))$ and taking the continuous-time limit $h \to 0$ this can be shown (Bronstein and Koeppl, 2018) to converge to the ODE in parameter space as

$$\frac{\mathrm{d}}{\mathrm{d}t}\theta(t) = F(\theta(t))^{-1}\, \mathsf{E}_{q(x\mid\theta(t))}\left[\mathcal{L}_t^{\dagger} \nabla_\theta \log q(X \mid \theta(t))\right], \tag{6}$$

where $F(\theta) := -\,\mathsf{E}_{q(x\mid\theta)}\left[\nabla_\theta \nabla_\theta^\top \log q(X \mid \theta)\right]$ is the Fisher information matrix of the parametric distribution $q(x \mid \theta)$. At the observation time points, we have discrete-time resets as

$$\theta(t_i) = \operatorname*{argmin}_{\theta'} \mathsf{KL}\big(p(y_i \mid x) q(x \mid \theta(t_i^-)) \parallel q(x \mid \theta')\big), \tag{7}$$

$\forall i = 1, \dots, N$, and the initial condition can be computed as

$$\theta(0) = \operatorname*{argmin}_{\theta'} \mathsf{KL}(p_0(x) \parallel q(x \mid \theta')). \tag{8}$$

The derivation is given for completeness in Appendix 2.1. Analogously, we use the entropic matching method backwards in time for the smoother $\tilde{\pi}_t(x) \approx q(x \mid \tilde{\theta}(t))$. Here, we compute for a small time step h the time-backwards update $\tilde{\theta}(t-h) = \operatorname{argmin}_{\tilde{\theta}} \mathsf{KL}(\tilde{\pi}_{t-h}(x) \parallel q(x \mid \tilde{\theta}))$. Considering again that $\tilde{\pi}_t(x) \approx q(x \mid \tilde{\theta}(t))$ and carrying out the continuous-time limit $h \to 0$ we derive the corresponding equation for the smoother as

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{\theta}(t) = -F(\tilde{\theta}(t))^{-1}\, \mathsf{E}_{q(x\mid\tilde{\theta}(t))}\left[\tilde{\mathcal{L}}_t^{\dagger} \nabla_\theta \log q(X \mid \tilde{\theta}(t))\right], \tag{9}$$

with end-point condition $\tilde{\theta}(T) = \theta(T)$. The full derivation is given in Appendix 2.2.

4.2 Expectation Propagation for Latent Markov Jump Processes

To further improve our approximation of the posterior, we can use the standard EP algorithm (Minka, 2001; Bishop, 2006; Wainwright et al., 2008; Cseke et al., 2016). We give here only a brief explanation of the algorithm, for a more detailed explanation we refer the reader to the provided resources. In the previously mentioned FFBS scheme, the contribution of the ith measurement to the posterior is included in the ith update of the filter as

$$\theta(t_i) = \theta(t_i^-) + \xi_i, \quad \forall i = 1, \dots, N, \tag{10}$$

with the likelihood contribution

$$\xi_i = \left(\operatorname*{argmin}_{\theta'} \mathsf{KL}\Big(p(y_i \mid x) q(x \mid \theta(t_i^-)) \,\Big\|\, q(x \mid \theta')\Big)\right) - \theta(t_i^-).$$

The EP algorithm optimizes the likelihood contributions ξ_i, also called site parameters, in an iterative manner. In each iteration j, the first step to update the site parameters is to calculate the *cavity parameters* $\tilde{\theta}_{\neg i}^{(j)}$, $\forall i = 1, \dots, N$, by excluding the ith likelihood contribution from the current approximate posterior estimate as

$$\tilde{\theta}_{\neg i}^{(j)} = \tilde{\theta}(t_i^-)^{(j)} - \xi_i^{(j)}, \quad \forall i = 1, \dots, N. \tag{11}$$

Next, the observation factors are incorporated to yield new approximate posterior parameters $\tilde{\theta}_{\setminus i}^{(j)}$ for the ith factor by computing the *tilted distribution* and projecting it back to the parametric family as

$$\tilde{\theta}_{\setminus i}^{(j)} = \operatorname*{argmin}_{\theta'} \mathsf{KL}\Big(p(y_i \mid x) q(x \mid \tilde{\theta}_{\neg i}^{(j)}) \,\Big\|\, q(x \mid \theta')\Big), \tag{12}$$

$\forall i = 1, \dots, N$. Subsequently, a revised site parameter is found such that, when combined with the cavity parameters, it yields the new approximate posterior. This is done by the subtraction of the cavity parameter as

$$\tilde{\xi}_i^{(j)} = \tilde{\theta}_{\setminus i}^{(j)} - \tilde{\theta}_{\neg i}^{(j)}, \quad \forall i = 1, \dots, N. \tag{13}$$

Finally, to have a convergent algorithm, a damped message passing strategy, with learning rate parameter $0 < \epsilon \leq 1$, is performed as

$$\xi_i^{(j+1)} = (1 - \epsilon)\xi_i^{(j)} + \epsilon \tilde{\xi}_i^{(j)}, \quad \forall i = 1, \dots, N. \tag{14}$$

The algorithm is then iterated by computing the posterior estimates $\tilde{\theta}(t_i^-)^{(j)}$, $\forall i = 1, \dots, N$, and subsequently the new site parameters $\xi_i^{(j+1)}$, $\forall i = 1, \dots, N$, for iteration steps $j = 1, 2, \dots$ until convergence. For initialization, we set the site parameters to zero, i.e., $\xi_1^{(1)} = 0, \dots, \xi_N^{(1)} = 0$, which corresponds to setting the initial approximate posterior to an approximate prior distribution. The EP algorithm can be shown to be a sensible algorithm, in the sense that it maximizes a tractable approximation of the intractable log marginal likelihood $\log p(y_{[0,T]})$ (Cseke et al., 2016) in the form of a fixed point equation for a relaxed variational principle of the intractable problem; for more information, see (Wainwright et al., 2008).

4.3 Parameter Learning

While our primary focus is on latent state inference, we utilize our approximate inference scheme to develop an approximate expectation maximization (EM) algorithm for parameter estimation, similar to the discrete-time case (Heskes et al., 2003). Therefore, we estimate the parameters ϕ of the system by maximizing a lower bound L of the marginal likelihood as

$$\begin{aligned} \log p(y_{[0,T]} \mid \phi) &\geq \mathrm{L}(\phi) \\ &:= \mathsf{E}_{\mathrm{Q}}\left[\log \frac{\mathrm{d}\,\mathrm{P}}{\mathrm{d}\,\mathrm{Q}}(X_{[0,T]}, y_{[0,T]}, \phi)\right], \end{aligned}$$

where $\frac{\mathrm{d}\,\mathrm{P}}{\mathrm{d}\,\mathrm{Q}}$ is the Radon-Nikodym derivative of the path measure $\mathrm{P}(X_{[0,T]} \in \cdot, Y_{[0,T]} \in \cdot \mid \phi)$ for the latent paths $X_{[0,T]} := \{X(t) \mid t \in [0,T]\}$ and all observations $Y_{[0,T]}$ w.r.t. an approximating probability measure $\mathrm{Q}(X_{[0,T]} \in \cdot)$ over latent paths and the Lebesgue measure over all observations $Y_{[0,T]}$. Note, that compared to the discrete-time case, we have to express this bound in terms of the Radon-Nikodym derivative, compared to an expression using probability densities w.r.t. the Lebesgue measure, since we have an uncountable number of random variables $\{X(t) \mid t \in [0,T]\}$ for which there is no Lebesgue measure, see, e.g., (Matthews et al., 2016). A standard result, see, e.g., (Bishop, 2006), is that the bound can be iteratively maximized by computing an expectation step (E-step) and a maximziation step (M-step). To compute the bound one has to exploit Girsanov's theorem (Kipnis and Landim, 1998; Hanson, 2007), but also other derivations can be found in the literature, e.g., (Opper and Sanguinetti, 2007; Cohn et al., 2010). We compute this bound in Appendix 3 in terms of the smoothing and filtering distribution, $\tilde{\pi}_t(x)$ and $\pi_t(x)$, respectively. Hence, by replacing these with the approximate smoothing and filtering distribution, $q(x \mid \tilde{\theta}(t))$ and $q(x \mid \theta(t))$, respectively, we can compute the M-step of the algorithm. This is achieved by maximizing the bound L which computes to

$$\begin{aligned} \mathrm{L}(\phi) = {} & \mathsf{E}_{q(x \mid \tilde{\theta}(0))}\left[\log \frac{p_0(X \mid \phi)}{q(X \mid \tilde{\theta}(0))}\right] + \int_0^T \mathsf{E}_{q(x \mid \tilde{\theta}(t))} \Bigg[\\ & \sum_{x' \neq X} \Bigg\{ \Lambda(X, x', t) \frac{q(x' \mid \tilde{\theta}(t))}{q(X \mid \tilde{\theta}(t))} \frac{q(X \mid \theta(t))}{q(x' \mid \theta(t))} \\ & \cdot \log \frac{\Lambda(X, x', t \mid \phi)}{\Lambda(X, x', t)} - \Lambda(X, x', t \mid \phi) \Bigg\} \Bigg] \mathrm{d}t \\ & + \sum_{i=1}^{N} \mathsf{E}_{q(x \mid \tilde{\theta}(t_i))}\left[\log p(y_i \mid X, \phi)\right] + C, \end{aligned} \tag{15}$$

where $\Lambda(x, x', t)$ are the rates of the approximating distribution Q independent of ϕ, $\Lambda(x, x', t \mid \phi)$ are the parameter dependent rates of the measure P and C is parameter independent constant. Therefore, the approximate EM algorithm can be achieved by carrying out the E-step, which consists of computing the approximate distributions $q(x \mid \theta(t))$ and $q(x \mid \tilde{\theta}(t))$ for all $t \in [0, T]$, keeping the parameters ϕ fixed. The M-step then consists of optimizing Eq. (15) w.r.t. the parameters ϕ, keeping the approximating distributions $q(x \mid \theta(t))$ and $q(x \mid \tilde{\theta}(t))$ fixed. For more details, see Appendix 3.

5 APPLICATION TO LATENT CHEMICAL REACTION NETWORKS

In this section, we apply the approximate inference algorithm to CRNs. Therefore, we consider systems, where the rate function can be expressed as in Eq. (2) with mass action kinetics as in Eq. (3). Throughout we will assume an exponential family form for the variational distribution as

$$q(x \mid \theta) = h(x) \exp(\theta^\top s(x) - A(\theta)),$$

with base measure $h : \mathcal{X} \to \mathbb{R}_{\geq 0}$, sufficient statistics $s : \mathcal{X} \to \mathbb{R}^p$ and log-partion function $A : \Theta \to \mathbb{R}$. Note, that $F(\theta) = \nabla_\theta \nabla_\theta^\top A(\theta)$ and $\nabla_\theta \log q(x \mid \theta) = s(x) - \mathsf{E}_{q(x \mid \theta)}[s(X)]$. As an instantiation we use a product

Poisson distribution

$$q(x \mid \theta) = \prod_{i=1}^{n} \mathrm{Pois}(x_i \mid \exp \theta_i).$$

Hence, we have base-measure $h(x) = \prod_{i=1}^{n} \frac{1}{x_i!}$, sufficient statistics $s(x) = x$, natural parameters or log rate parameters $\theta = [\theta_1, \ldots, \theta_n]^\top$ and log-partition function $A(\theta) = \sum_{i=1}^{n} \exp(\theta_i)$. For the measurements we assume a linear Gaussian measurement model for the latent state, i.e., we assume the observation likelihood $p(y_i \mid x) = \mathcal{N}(y_i \mid Hx, \Sigma)$, with the ith observation $y_i \in \mathbb{R}^m$, the observation model matrix $H \in \mathbb{R}^{m \times n}$, and observation noise covariance $\Sigma \in \mathbb{R}^{m \times m}$. Note, that the following derivations can also be applied to general non-linear observation models $H(x)$, by following a linearization procedure, as in extended Kalman filtering approaches, see, e.g., (Särkkä, 2013).

5.1 Approximate Filtering and Smoothing

For approximate inference, we first consider the initialization step, see Eq. (8). Here, we assume that $p_0(x) = \prod_{i=1}^{n} \mathrm{Pois}(x_i \mid \exp(\theta_{i,0}))$, with given initial log rate parameters $\{\theta_{1,0}, \ldots, \theta_{n,0}\}$. Hence, Eq. (8) computes to the initial parameter $\theta(0) = [\theta_{1,0}, \ldots, \theta_{n,0}]^\top$. Next, we compute the prediction step for the filtering distribution in Eq. (6). For CRNs, the product Poisson variational distribution leads to closed-form updates, see Appendix 4.1, for the prior drift as

$$\frac{\mathrm{d}}{\mathrm{d}t}\theta(t) = F(\theta(t))^{-1} \sum_{j=1}^{k} c_j \nu_j \exp\left(\sum_{i=1}^{n} \underline{\nu}_{ij} \theta_i(t) \right), \quad (16)$$

with the inverse Fisher information matrix $F(\theta)^{-1} = \mathrm{diag}([\exp(-\theta_1), \ldots, \exp(-\theta_n)]^\top)$.

At the observation time points, we compute the update from $\theta = \theta(t_i^-)$ to $\theta^* = \theta(t)$ according to Eq. (7). Given the exponential family distribution for $q(x \mid \theta)$ this leads to a moment matching condition as $\mathsf{E}_{q(x \mid \theta^*)}[s(X)] = \mathsf{E}_{p(x \mid y_i)}[s(X)]$, where $p(x \mid y_i)$ is the exact posterior distribution. Since the Gaussian likelihood $p(y_i \mid x)$ is not conjugate to the product Poisson distribution $q(x \mid \theta)$, the computation of the exact posterior $p(x \mid y_i)$ is generally intractable. We propose an additional moment matching scheme in order to get closed-form updates, see Appendix 4.2. The resulting mean of the posterior can then be written as

$$m = \exp \theta + F(\theta) H^\top (H F(\theta) H^\top + \Sigma)^{-1} (y_i - H \exp \theta),$$

where the exponential function operating on vectors is applied component wise. Note that the posterior mean can be negative, as such we truncate it at a small value $0 < \epsilon_\lambda \ll 1$. Hence, the full approximate update in the natural parameter space can be written as

$$\xi_i = \log\left(\max\left\{m, \epsilon_\lambda\right\}\right) - \theta, \quad (17)$$

where we set $\epsilon_\lambda = 10^{-6}$.

Given the filtering distribution we can then find a closed form expression for the backward smoothing step in Eq. (9) as

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{\theta}(t) = F(\tilde{\theta}(t))^{-1} \sum_{j=1}^{k} c_j \nu_j \exp\left(\sum_{i=1}^{n} \underline{\nu}_{ij} \tilde{\theta}_i(t) \right) \cdot \exp\left(\sum_{i=1}^{n} \nu_{ij} (\tilde{\theta}_i(t) - \theta_i(t)) \right), \quad (18)$$

for the derivation see Appendix 4.3.

5.2 Expectation Propagation for Latent Chemical Reaction Networks

Finally, for EP we update the site parameters as described in Eq. (13). Specifically, $\tilde{\xi}_i^{(j)}$ is obtained from Eq. (17) by substituting $\theta = \tilde{\theta}_{\neg i}^{(j)}$, the cavity parameters computed using Eq. (11). This step corresponds to computing the tilted distribution, projecting it back to the parametric family, and subtracting the cavity parameters. We summarize our method in Algorithm 1.

```
Input: Time Horizon $T$, observations $\{y_1, \ldots, y_N\}$, observation times $\{t_1, \ldots, t_N\}$, initial parameters $\theta(0)$, EP iterations $K$, learning rate parameter $\epsilon$
Output: Smoother parameter $\tilde{\theta}$, site parameter $\xi_i$
Initialize sites $\xi_i = 0$
for $j \leftarrow 1$ to $K$ do
    // Compute the filter
    Initialize time $t = 0$
    for $i \leftarrow 1$ to $N$ do
        Solve filter ODE (16) from $t$ to $t_i$ with initial condition $\theta(t)$
        Update the filter at $\theta(t_i)$ via Eq. (10)
        Update time $t \leftarrow t_i$
    end
    Solve filter ODE (16) from $t_N$ to $T$ with initial condition $\theta(t)$
    // Compute the smoother
    Solve smoother ODE (18) from $T$ to 0 with initial condition $\tilde{\theta}(T) = \theta(T)$
    // Expectation Propagation Steps
    Compute the cavity parameter $\tilde{\theta}_{\neg i}$ via Eq. (11)
    Compute the new site parameters $\tilde{\xi}_i$ via Eq. (17) by setting $\theta = \tilde{\theta}_{\neg i}$
    Perform a damped update of the site parameters $\xi_i$ via Eq. (14)
end
```

Algorithm 1: Expectation propagation for CRNs

5.3 Parameter Learning

For the given approximation of a CRN model, we can find closed form solutions for the M-step of the EM algorithm. Here, we consider the initial parameters $\{\theta_{1,0}, \ldots, \theta_{n,0}\}$, the effective rate parameters $\{c_1, \ldots, c_k\}$ and the observation model parameters H and Σ, i.e., $\phi = \{\{\theta_{i,0}\}_{i=1}^n, \{c_j\}_{j=1}^k, H, \Sigma\}$. In Appendix 5 we show how to compute the bound in Eq. (15) w.r.t. the individual parameters ϕ in closed form. Additionally, we derive closed-form coordinate-wise update schemes for the individual parameters ϕ.

6 EXPERIMENTS

We evaluate our proposed method across various instantiations of CRNs and compare the results against multiple baseline approaches. In this section, we highlight the latent state inference tasks for both a Lotka-Volterra model and a motility model. Detailed information on the experiments, additional benchmarks, and insights into the parameter learning task can be found in Appendix 6.

Lotka-Volterra Model. The Lotka-Volterra or predator-prey model, see, e.g., (Wilkinson, 2018), is one of the most well-known models in population dynamics. It describes the dynamic evolution of a prey species X_1 and a predator species X_2. The model can be described by three reactions as

$$\mathsf{X}_1 \xrightarrow{c_1} 2\mathsf{X}_1, \quad \mathsf{X}_1 + \mathsf{X}_2 \xrightarrow{c_2} 2\mathsf{X}_2, \qquad \mathsf{X}_2 \xrightarrow{c_3} \emptyset.$$

These reactions reflect the prey growth, the predator reproduction and the predator decline, with rates c_1, c_2, and c_3, respectively. We model the reaction network system using an MJP with mass-action kinetics as in Eqs. (2) and (3). We simulate the system using the Doob-Gillespie algorithm (Doob, 1945; Gillespie, 1976) and consider that both species are observed at non-equidistant discrete time points subject to independent Gaussian noise. For inference, we exploit the discussed entropic matching method with EP, using a product Poisson approximation.

We evaluate the performance on inferring the latent state path $X_{[0,T]}$ conditioned on the observations $Y_1, \ldots, Y_N$. Fig. 2 depicts the qualitative results of our method for a sample path with 10 observations. We notice that the inferred posterior mean closely tracks the qualitative behavior of the system dynamics. Additionally, the Poisson approximation gives a measure of uncertainty, indicated in the plot as the background.

For quantitative analysis, we infer the approximate posterior for 100 sample trajectories and compare our results to various baselines, namely: (i) a single FFBS iteration using the entropic matching method without EP, (ii) a Gaussian assumed density smoother (ADS) based on the chemical Langevin equation, similar to the method described by Cseke et al. (2016), (iii) the moment-based VI method proposed by Wildner and Koeppl (2019), and (iv) an exact smoothing algorithm based on a truncation of the system, which serves as ground truth for our comparison. Table 1 shows the mean squared error in the posterior mean of the approximate methods compared to the exact posterior mean of the truncated system. Our method demonstrates superior performance overall. The improvement achieved with the EP algorithm is substantial compared to the single FFBS iteration with entropic matching. Our method also outperforms the Gaussian ADS, which suffers from approximation errors due to modeling the MJP as an SDE. This underscores the benefit of directly modeling the system as an MJP, especially for low populations. Finally, the moment-based VI method optimizes the rates describing the posterior process, whereas our method optimizes the parameters of the posterior marginal distributions directly, which leads to better mean estimates.

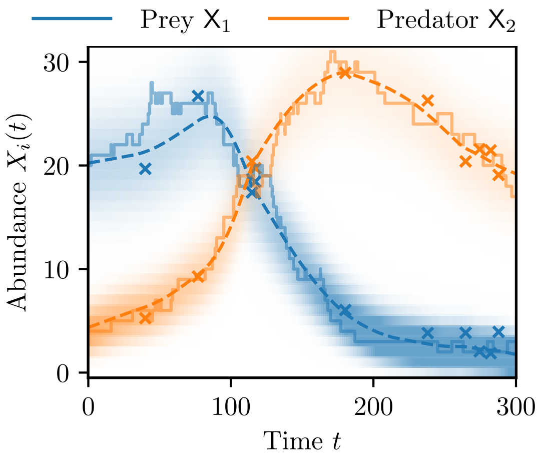

Figure 2: Simulation and latent state inference of a Lotka-Volterra model. Solid lines represent the ground truth trajectory, while crosses indicate the observations. Approximate inference results of our method are visualized with dashed lines for the posterior mean using and the background indicates the inferred marginal state probabilities.

Table 1: Mean squared error in posterior mean averaged over trajectories and time

EP (Ours)	FFBS Entropic	G ADS	MBVI
0.4581	2.1989	1.9671	1.6951

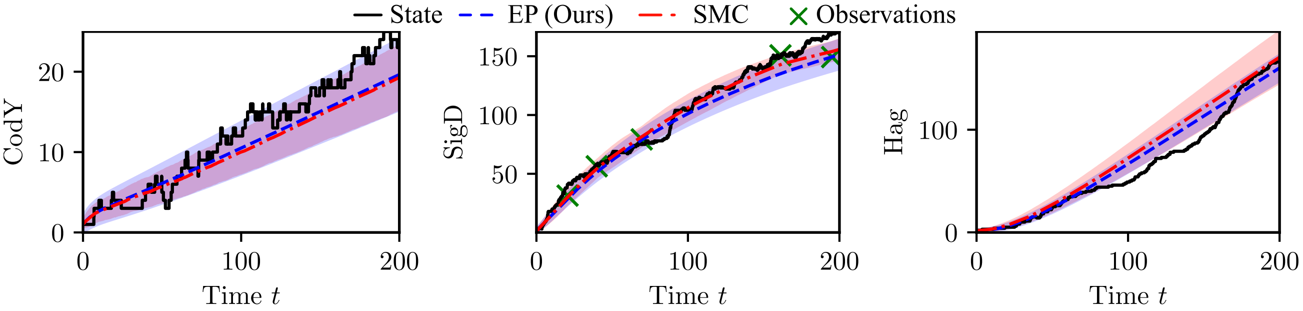

Figure 3: Three selected species from the motility model adapted from (Wilkinson, 2010), where only the species SigD is noisily observed (crosses). The plots compare the results of our method and an SMC approach by indicating their resulting mean and standard deviation.

Motility Model. As a second more challenging example, we consider the motility model for bacterial gene regulation introduced by Wilkinson (2010). The model consists of a total of nine species and twelve reactions as

$$\text{codY} \xrightarrow{c_1} \text{codY} + \text{CodY}, \quad \text{CodY} \xrightarrow{c_2} \emptyset,$$
$$\text{flache} \xrightarrow{c_3} \text{flache} + \text{SigD}, \quad \text{SigD} \xrightarrow{c_4} \emptyset,$$
$$\text{SigD\_hag} \xrightarrow{c_5} \text{SigD} + \text{hag} + \text{Hag}, \quad \text{Hag} \xrightarrow{c_6} \emptyset,$$
$$\text{SigD} + \text{hag} \xrightarrow{c_7} \text{SigD\_hag}, \quad \text{SigD\_hag} \xrightarrow{c_8} \text{SigD} + \text{hag},$$
$$\text{CodY} + \text{flache} \xrightarrow{c_9} \text{CodY\_flache},$$
$$\text{CodY\_flache} \xrightarrow{c_{10}} \text{CodY} + \text{flache},$$
$$\text{CodY} + \text{hag} \xrightarrow{c_{11}} \text{CodY\_hag},$$
$$\text{CodY\_hag} \xrightarrow{c_{12}} \text{CodY} + \text{hag}.$$

In this setup, we only have access to noisy observations of SigD. The exact smoothing algorithm based on truncating the state space, used as ground truth in the Lotka-Volterra model, becomes intractable for higher-dimensional models like the motility model due to the exponential growth of the state space. This limitation underscores the necessity for approximate methods. We again exploit a product Poisson approximation for the posterior distribution and estimate the latent state. To validate our approach we compare the results with an SMC approach, with a sufficiently large number of samples $N_s = 10000$, which acts as a reliable approximation of the ground truth. We did not compare to the Gaussian ADS due to numerical instability, nor to the moment-based VI method, as the number of moment equations that must be manually derived grows quadratically with the number of species. Fig. 3 shows qualitative results for three of the nine species given a sample trajectory. The additional trajectories can be found in Appendix 6.

We observe that the trajectories for the noisily observed SigD, as well as the fully latent Hag and CodY, can be tracked accurately. Our approximate solution yields results that are closely aligned with those obtained using the SMC approach, which serves as the ground truth. This demonstrates that our method provides reliable latent state inference while offering a more practical and scalable alternative for complex models.

7 CONCLUSION

We presented a principled inference framework for MJPs based on an entropic matching method embedded into an EP algorithm. The new method arrives at closed-form results for the important class of CRNs used within the field of systems biology. The analytic nature of the closed-form message passing scheme makes the method highly scalable and fast.

One limitation of our work is the expressiveness of the product Poisson approximation. As it only supplies one degree of freedom per species, it can only model an increase in variance by an increase in mean. Nevertheless, even with this limited variational distribution, our method demonstrates strong performance. This highlights the potential of our approach for addressing complex continuous-time Bayesian inference problems. Future work can build on this foundation by exploring more expressive variational distributions to further enhance the accuracy and applicability of the method. A potential candidate could be an energy based model (Du and Mordatch, 2019), which would yield an expressive class for the approximate posterior distribution. This would go well beyond the presented setup of using a product form posterior distribution with only one parameter per dimension. Though closed-form analytic updates might not be possible anymore, we think that advanced MCMC methods (Sun et al., 2023) could still lead to a scalable algorithm. This would enable for rather complex observation likelihoods, which have for example also been discussed in the context of discrete-time systems (Johnson et al., 2016) and within the EP framework (Vehtari et al., 2020). Moreover, we aim to extend our framework to other classes of MJPs, such as queueing systems, where entropic matching has recently been used for the filtering problem (Eich et al., 2024).

Acknowledgements

This work has been co-funded by the German Research Foundation (DFG) within the Collaborative Research Center (CRC) 1053 MAKI and project number 517777863 and by the Federal Ministry of Education and Research as part of the Software Campus project RL4MFRP (funding code 01IS23067).

References

B. D. Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12 (3):313–326, 1982.

B. D. Anderson and I. B. Rhodes. Smoothing algorithms for nonlinear finite-dimensional systems. *Stochastics: An International Journal of Probability and Stochastic Processes*, 9(1-2):139–165, 1983.

D. F. Anderson and T. G. Kurtz. *Stochastic analysis of biochemical systems*. Springer, 2015.

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

C. Archambeau and M. Opper. Approximate inference for continuous-time Markov processes. *Bayesian Time Series Models*, pages 125–140, 2011.

C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

G. Bolch, S. Greiner, H. De Meer, and K. S. Trivedi. *Queueing networks and Markov chains: modeling and performance evaluation with computer science applications*. John Wiley & Sons, 2006.

L. Bronstein and H. Koeppl. A variational approach to moment-closure approximations for the kinetics of biomolecular reaction networks. *The Journal of chemical physics*, 148(1):014105, 2018.

I. Cohn, T. El-Hay, N. Friedman, and R. Kupferman. Mean field variational approximation for continuous-time Bayesian networks. *The Journal of Machine Learning Research*, 11:2745–2783, 2010.

B. Cseke, D. Schnoerr, M. Opper, and G. Sanguinetti. Expectation propagation for continuous time stochastic processes. *Journal of Physics A: Mathematical and Theoretical*, 49(49), 2016.

P. Del Moral and S. Penev. *Stochastic Processes: From Applications to Theory*. Chapman and Hall/CRC, 2017.

J. L. Doob. Markoff chains—denumerable case. *Transactions of the American Mathematical Society*, 58: 455–473, 1945.

A. Doucet, N. de Freitas, and N. J. Gordon. *Sequential Monte Carlo methods in practice*, volume 1. Springer, 2001.

Y. Du and I. Mordatch. Implicit generation and modeling with energy based models. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Y. Eich, B. Alt, and H. Koeppl. Approximate control for continuous-time POMDPs. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, pages 3160–3168. PMLR, 2024.

R. Elliott. Reverse-time Markov processes (corresp.). *IEEE Transactions on Information Theory*, 32(2): 290–292, 1986.

S. N. Ethier and T. G. Kurtz. *Markov processes: characterization and convergence*. John Wiley & Sons, 2009.

C. W. Gardiner et al. *Handbook of stochastic methods*, volume 3. Springer Berlin, 1985.

D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of computational physics*, 22(4):403–434, 1976.

D. T. Gillespie. The chemical Langevin equation. *The Journal of Chemical Physics*, 113(1):297–306, 2000.

A. Golightly and D. J. Wilkinson. Bayesian parameter inference for stochastic biochemical network models using particle Markov chain Monte Carlo. *Interface Focus*, 1(6):807–820, 2011.

A. Golightly, D. A. Henderson, and C. Sherlock. Delayed acceptance particle mcmc for exact inference in stochastic kinetic models. *Statistics and Computing*, 25:1039–1055, 2015.

F. B. Hanson. *Applied stochastic processes and control for jump-diffusions: Modeling, analysis and computation*. SIAM, 2007.

T. Heskes, O. Zoeter, and W. Wiegerinck. Approximate expectation maximization. In *Advances in Neural Information Processing Systems*, volume 16, 2003.

L. Huang, L. Pauleve, C. Zechner, M. Unger, A. S. Hansen, and H. Koeppl. Reconstructing dynamic molecular states from single-cell time series. *Journal of The Royal Society Interface*, 13(122), 2016.

M. J. Johnson, D. K. Duvenaud, A. Wiltschko, R. P. Adams, and S. R. Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

C. Kipnis and C. Landim. *Scaling limits of interacting particle systems*, volume 320. Springer Science & Business Media, 1998.

M. Komorowski, B. Finkenstädt, C. V. Harper, and D. A. Rand. Bayesian inference of biochemical kinetic parameters using the linear noise approximation. *BMC Bioinformatics*, 10:1–10, 2009.

T. E. Lowe, A. Golightly, and C. Sherlock. Accelerating inference for stochastic kinetic models. *Computational Statistics & Data Analysis*, 185:107760, 2023.

R. S. Mamon and R. J. Elliott. *Hidden Markov models in finance*, volume 4. Springer, 2007.

A. G. d. G. Matthews, J. Hensman, R. Turner, and Z. Ghahramani. On sparse variational methods and the Kullback-Leibler divergence between stochastic processes. In *Artificial Intelligence and Statistics*, pages 231–239. PMLR, 2016.

P. S. Maybeck. *Stochastic models, estimation, and control*. Academic Press, 1982.

M. Mider, M. Schauer, and F. Van der Meulen. Continuous-discrete smoothing of diffusions. *Electronic Journal of Statistics*, 15(2):4295–4342, 2021.

T. P. Minka. *A family of algorithms for approximate Bayesian inference*. PhD thesis, Massachusetts Institute of Technology, 2001.

J. R. Norris. *Markov chains*. Cambridge University Press, 1998.

M. Opper and G. Sanguinetti. Variational inference for Markov jump processes. In *Advances in Neural Information Processing Systems*, volume 20, 2007.

E. Pardoux. Non-linear filtering, prediction and smoothing. In *Stochastic Systems: The Mathematics of Filtering and Identification and Applications: Proceedings of the NATO Advanced Study Institute held at Les Arcs, Savoie, France, June 22–July 5, 1980*, pages 529–557. Springer, 1981.

T. Ramalho, M. Selig, U. Gerland, and T. A. Enßlin. Simulation of stochastic network dynamics via entropic matching. *Phys. Rev. E*, 87, 2013.

S. Särkkä. *Bayesian filtering and smoothing*. Cambridge University Press, 2013.

S. Särkkä and A. Solin. *Applied stochastic differential equations*. Cambridge University Press, 2019.

P. Seifner and R. J. Sánchez. Neural markov jump processes. In *International Conference on Machine Learning*, pages 30523–30552. PMLR, 2023.

H. Sun, H. Dai, B. Dai, H. Zhou, and D. Schuurmans. Discrete Langevin samplers via wasserstein gradient flow. In *International Conference on Artificial Intelligence and Statistics*, pages 6290–6313. PMLR, 2023.

R. Van Handel. *Filtering, stability, and robustness*. PhD thesis, California Institute of Technology, 2007.

A. Vehtari, A. Gelman, T. Sivula, P. Jylänki, D. Tran, S. Sahai, P. Blomstedt, J. P. Cunningham, D. Schiminovich, and C. P. Robert. Expectation propagation as a way of life: A framework for Bayesian inference on partitioned data. *The Journal of Machine Learning Research*, 21(1):577–629, 2020.

M. J. Wainwright, M. I. Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1 (1–2):1–305, 2008.

C. Wildner and H. Koeppl. Moment-based variational inference for Markov jump processes. In *International Conference on Machine Learning*, pages 6766–6775. PMLR, 2019.

D. J. Wilkinson. Parameter inference for stochastic kinetic models of bacterial gene regulation: a Bayesian approach to systems biology. In *Proceedings of 9th Valencia International Meeting on Bayesian Statistics*, pages 679–705, 2010.

D. J. Wilkinson. *Stochastic modelling for systems biology*. Chapman and Hall/CRC, 2018.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes, see additional information on experiments in Appendix 6]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]
2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]
3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]

(b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes, see Appendix 6]

(c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]

(d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes, see Appendix 6]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:

   (a) Citations of the creator If your work uses existing assets. [Not Applicable]

   (b) The license information of the assets, if applicable. [Not Applicable]

   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]

   (d) Information about consent from data providers/curators. [Not Applicable]

   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

   (a) The full text of instructions given to participants and screenshots. [Not Applicable]

   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Entropic Matching for Expectation Propagation of Markov Jump Processes: Supplementary Materials

1 SOME NOTES ON EXACT FILTERING AND SMOOTHING

To find the smoothing distribution $\tilde{\pi}_t(x)$ we have multiple options. It can be shown Huang et al. (2016); Anderson and Rhodes (1983); Pardoux (1981) that the smoothing distribution can be factorized as

$$\tilde{\pi}_t(x) = \frac{\pi_t(x)\beta_t(x)}{\sum_{x' \in \mathcal{X}} \pi_t(x')\beta_t(x')}, \tag{19}$$

where $\beta_t(x)$ is the backward-filtering distribution given by $\beta_t(x) \coloneqq p(y_{(t,T]} \mid X(t) = x)$, where $y_{(t,T]} \coloneqq \{y_i \mid i \in \{1, \dots, N\} : t_i \in (t, T]\}$ are only the "future" observations. Note that $\tilde{\pi}_T(x) = \mathrm{P}(X(T) = x \mid y_{[0,T]}) = \pi_T(x)$, hence we have $\beta_T(x) = 1$. The backward-filtering distribution in between observation time points follows the differential form of the backward Chapman-Kolmogorov equation Gardiner et al. (1985) and is updated at observation time points Huang et al. (2016), hence, we have

$$\beta_T(x) = 1, \quad \frac{\mathrm{d}}{\mathrm{d}t}\beta_t(x) = -[\mathcal{L}_t^\dagger \beta_t](x), \qquad \beta_{t_i^-}(x) = \frac{p(y_i \mid x)\beta_{t_i}(x)}{\sum_{x' \in \mathcal{X}} p(y_i \mid x')\beta_{t_i}(x')}.$$

Note that $\beta_t(x)$ is càglàd in t. Therefore, we can compute the smoothing distribution $\tilde{\pi}_t(x)$, by first computing the filtering distribution $\pi_t(x)$ forwardly in time, subsequently, computing the backward-filtering distribution $\beta_t(x)$ backwardly in time and then use Eq. (19) to compute $\tilde{\pi}_t(x)$.

Another way is to note that an initial condition for the smoothing distribution is given by $\tilde{\pi}_0(x) \propto p_0(x)\beta_0(x)$. Differentiating Eq. (19) by time, then gives rise to the forward smoothing dynamics Huang et al. (2016); Anderson and Rhodes (1983)

$$\tilde{\pi}_0(x) = \frac{p_0(x)\beta_0(x)}{\sum_{x' \in \mathcal{X}} p_0(x')\beta_0(x')}, \quad \frac{\mathrm{d}}{\mathrm{d}t}\tilde{\pi}_t(x) = [\bar{\mathcal{L}}_t \tilde{\pi}_t](x), \tag{20}$$

where the forward smoothing operator $\bar{\mathcal{L}}_t$ for an arbitrary test function ϕ and the forward smoothing rate function $\bar{\Lambda}(x, x', t)$ are given as

$$[\bar{\mathcal{L}}_t \phi](x) \coloneqq \sum_{x' \neq x} \bar{\Lambda}(x', x, t)\phi(x') - \bar{\Lambda}(x, x', t)\phi(x), \quad \bar{\Lambda}(x, x', t) = \Lambda(x, x', t)\frac{\beta_t(x')}{\beta_t(x)}. \tag{21}$$

Hence, the smoothing distribution $\tilde{\pi}_t(x)$ can also be calculated by solving the backwards filtering distribution $\beta_t(x)$ backwardly in time and subsequently solving for the smoothing distribution $\tilde{\pi}_t(x)$ using the forward smoothing dynamics in Eq. (20).

2 DERIVATION FOR THE FILTERING AND SMOOTHING VARIATIONAL PARAMETERS

2.1 Entropic Matching for Filtering

Here, we find the variational parameters $\theta(t)$ by considering

$$\theta(t+h) = \operatorname*{argmin}_{\theta'} \mathsf{KL}(\pi_{t+h}(x) \parallel q(x \mid \theta')),$$

where $\pi_{t+h}(x)$ is the filtering distribution that is propagated for a small time step h. We get an recursive algorithm by considering that the filtering dsitrbution at time point t has the initial distribution $q(x \mid \theta(t))$, i.e., we have

$$\frac{\mathrm{d}}{\mathrm{d}t}\pi_t(x) = [\mathcal{L}_t \pi_t](x), \quad \pi_t(x) = q(x \mid \theta(t)).$$

Hence, we can write

$$\pi_{t+h}(x) = q(x \mid \theta(t)) + h\mathcal{L}_t q(x \mid \theta(t)) + o(h),$$

with $\lim_{h\to 0} {}^{o(h)}/_{h} = 0$. For the KL divergence we compute

$$\begin{aligned}
&\mathsf{KL}(\pi_{t+h}(x) \parallel q(x \mid \theta')) \\
&= \mathsf{KL}(q(x \mid \theta(t)) + h\mathcal{L}_t q(x \mid \theta(t)) + o(h) \parallel q(x \mid \theta')) \\
&= \mathsf{E}_{\pi_{t+h}(x)}\left[\log \frac{q(X \mid \theta(t)) + h\mathcal{L}_t q(X \mid \theta(t)) + o(h)}{q(X \mid \theta')}\right] \\
&= \mathsf{E}_{\pi_{t+h}(x)}\left[\log\left(q(X \mid \theta(t)) + h\mathcal{L}_t q(X \mid \theta(t)) + o(h)\right) - \log q(X \mid \theta')\right]
\end{aligned}$$

Using a Taylor series around $h = 0$, we can find that for some coefficients a and b, we have

$$\log(a + bh + o(h)) = \log a + h\frac{b}{a} + o(h).$$

Hence, we calculate

$$\begin{aligned}
&\mathsf{KL}(\pi_{t+h}(x) \parallel q(x \mid \theta')) \\
&\quad= \mathsf{E}_{\pi_{t+h}(x)}\left[\log\left(q(X \mid \theta(t)) + h\mathcal{L}_t q(X \mid \theta(t)) + o(h)\right) - \log q(X \mid \theta')\right] \\
&\quad= \mathsf{E}_{\pi_{t+h}(x)}\left[\log q(X \mid \theta(t) + h\frac{\mathcal{L}_t q(X \mid \theta(t))}{q(X \mid \theta(t)} + o(h) - \log q(X \mid \theta')\right] \\
&\quad= \sum_{x\in\mathcal{X}} \pi_{t+h}(x)\left[\log q(x \mid \theta(t) + h\frac{\mathcal{L}_t q(x \mid \theta(t))}{q(x \mid \theta(t)} + o(h) - \log q(x \mid \theta')\right] \\
&\quad= \sum_{x\in\mathcal{X}} \left(q(x \mid \theta(t)) + h\mathcal{L}_t q(x \mid \theta(t)) + o(h)\right) \\
&\qquad\qquad\qquad \cdot \left[\log q(x \mid \theta(t) + h\frac{\mathcal{L}_t q(x \mid \theta(t))}{q(x \mid \theta(t)} + o(h) - \log q(x \mid \theta')\right] \\
&\quad= \sum_{x\in\mathcal{X}} \Bigg\{ q(x \mid \theta(t)) \log\frac{q(x \mid \theta(t))}{q(x \mid \theta')} \\
&\qquad\quad + h\left[q(x \mid \theta(t))\frac{\mathcal{L}_t q(x \mid \theta(t))}{q(x \mid \theta(t))} + \mathcal{L}_t q(x \mid \theta(t)) \log\frac{q(x \mid \theta(t))}{q(x \mid \theta')}\right] + o(h)\Bigg\} \\
&\quad= \mathsf{KL}(q(x \mid \theta(t)) \parallel q(x \mid \theta')) \\
&\qquad + h\,\mathsf{E}_{q(x\mid\theta(t))}\left[\frac{\mathcal{L}_t q(X \mid \theta(t))}{q(X \mid \theta(t)} + \frac{\mathcal{L}_t q(X \mid \theta(t))}{q(X \mid \theta(t))}\log\frac{q(X \mid \theta(t))}{q(X \mid \theta')}\right] + o(h).
\end{aligned}$$

By assuming that for small h the parameter $\theta' = \theta(t+h)$ is close to the parameter $\theta = \theta(t)$, we can exploit a series expansion in $\theta - \theta'$ up to second order of the KL divergence as

$$\mathsf{KL}(q(x \mid \theta) \parallel q(x \mid \theta')) = \frac{1}{2}(\theta' - \theta)^\top F(\theta)(\theta' - \theta),$$

where $F(\theta) \coloneqq -\,\mathsf{E}_{q(x\mid\theta)}\left[\nabla_\theta \nabla_\theta^\top \log q(X \mid \theta)\right]$ is the Fisher information matrix. Hence, we compute

$$\begin{aligned}0 &= \nabla_{\theta'}\,\mathsf{KL}(\pi_{t+h}(x) \parallel q(x \mid \theta'))|_{\theta'=\theta(t+h)}\\ &= F(\theta(t))(\theta(t+h)-\theta(t)) - h\,\mathsf{E}_{q(x\mid\theta(t))}\left[\nabla_{\theta'} \log q(X \mid \theta(t+h)) \frac{\mathcal{L}_t q(X \mid \theta(t))}{q(X \mid \theta(t))}\right] + o(h)\end{aligned}$$

Dividing both sides by h and taking the limit $h \to 0$ we obtain

$$\begin{aligned}0 &= F(\theta(t))\frac{\mathrm{d}}{\mathrm{d}t}\theta(t) - \mathsf{E}_{q(x\mid\theta(t))}\left[\nabla_\theta \log q(X \mid \theta(t)) \frac{\mathcal{L}_t q(X \mid \theta(t))}{q(X \mid \theta(t))}\right]\\ &= F(\theta(t))\frac{\mathrm{d}}{\mathrm{d}t}\theta(t) - \sum_x q(x \mid \theta(t)) \nabla_\theta \log q(x \mid \theta(t)) \frac{\mathcal{L}_t q(x \mid \theta(t))}{q(x \mid \theta(t))}\\ &= F(\theta(t))\frac{\mathrm{d}}{\mathrm{d}t}\theta(t) - \sum_x \nabla_\theta \log q(x \mid \theta(t)) \mathcal{L}_t q(x \mid \theta(t))\\ &= F(\theta(t))\frac{\mathrm{d}}{\mathrm{d}t}\theta(t) - \sum_x q(x \mid \theta(t)) \mathcal{L}_t^\dagger \nabla_\theta \log q(x \mid \theta(t))\\ &= F(\theta(t))\frac{\mathrm{d}}{\mathrm{d}t}\theta(t) - \mathsf{E}_{q(x\mid\theta(t))}\left[\mathcal{L}_t^\dagger \nabla_\theta \log q(X \mid \theta(t))\right].\end{aligned}$$

This leads to the drift in between observation time points

$$\frac{\mathrm{d}}{\mathrm{d}t}\theta(t) = F(\theta(t))^{-1}\,\mathsf{E}_{q(x\mid\theta(t))}\left[\mathcal{L}_t^\dagger \nabla_\theta \log q(X \mid \theta(t))\right].$$

At the observation time points we reset the parameter to

$$\theta(t_i) = \operatorname*{argmin}_{\theta'} \mathsf{KL}\big(Z^{-1} p(y_i \mid x) q(x \mid \theta(t_i^-)) \parallel q(x \mid \theta')\big),$$

where the first argument in the KL divergence corresponds to the posterior distribution obtained from Bayes' rule and Z refers to the normalization constant. We usually use the unnormalized posterior as first argument, which does not affect the optimization, as the normalization constant does not depend on θ'.

The initial condition $\theta(0) = \theta_0$ is given by

$$\theta_0 = \operatorname*{argmin}_{\theta'} \mathsf{KL}(p_0(x) \parallel q(x \mid \theta')).$$

2.2 Entropic Matching for Smoothing

Similar to the filtering approximation, we use the entropic matching method backwards in time for the smoother

$$\tilde{\pi}_t(x) \approx q(x \mid \tilde{\theta}(t)).$$

Here, we compute

$$\tilde{\theta}(t-h) = \operatorname*{argmin}_{\tilde{\theta}} \mathsf{KL}\Big(\tilde{\pi}_{t-h}(x) \,\Big\Vert\, q(x \mid \tilde{\theta})\Big).$$

Further, we assume that

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{\pi}_t(x) = -\tilde{\mathcal{L}}_t \tilde{\pi}_t(x), \quad \tilde{\pi}_t(x) = q(x \mid \tilde{\theta}(t)).$$

For the smoothing distribution we can compute the evolution backwards in time by

$$\tilde{\pi}_{t-h}(x) = q(x \mid \tilde{\theta}(t)) + h\tilde{\mathcal{L}}_t q(x \mid \tilde{\theta}(t)) + o(h).$$

Hence, similar to the derivation of the approximate filter we have

$$0 = F(\tilde{\theta}(t))(\tilde{\theta}(t-h) - \tilde{\theta}(t)) - h\,\mathsf{E}_{q(x\mid\tilde{\theta}(t))}\left[\nabla_{\tilde{\theta}} \log q(X \mid \tilde{\theta}(t-h)) \frac{\tilde{\mathcal{L}}_t q(X \mid \tilde{\theta}(t))}{q(X \mid \tilde{\theta}(t))}\right] + o(h).$$

Dividing both sides by h and taking the limit $h \to 0$ we arrive analog to the above derivation at

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{\theta}(t) = -F(\tilde{\theta}(t))^{-1}\,\mathsf{E}_{q(x\mid\tilde{\theta}(t))}\left[\tilde{\mathcal{L}}_t^\dagger \nabla_\theta \log q(X \mid \tilde{\theta}(t))\right],$$

with end-point condition $\tilde{\theta}(T) = \theta(T)$.

3 PARAMETER LEARNING

For parameter learning we use an approximate EM scheme. For the derivation of the EM algorithm, we consider that instead of maximizing the intractable marginal likelihood $p(y_{[0,T]} \mid \phi)$ w.r.t. some parameters ϕ, a tractable lower bound $\mathrm{L}(\phi)$ as

$$\log p(y_{[0,T]} \mid \phi) \geq \mathrm{L}(\phi) \coloneqq \mathsf{E}_{\mathrm{Q}}\left[\log \frac{\mathrm{d}\,\mathrm{P}}{\mathrm{d}\,\mathrm{Q}}(X_{[0,T]}, y_{[0,T]}, \phi)\right],$$

where $\frac{\mathrm{d}\,\mathrm{P}}{\mathrm{d}\,\mathrm{Q}}$ is the Radon-Nikodym derivative of the path measure $\mathrm{P}(X_{[0,T]} \in \cdot, Y_{[0,T]} \in \cdot \mid \phi)$ for the latent paths $X_{[0,T]} \coloneqq \{X(t) \mid t \in [0,T]\}$ and all observations $Y_{[0,T]}$ w.r.t. an an approximating probability measure $\mathrm{Q}(X_{[0,T]} \in \cdot)$ over latent paths and the Lebesgue measure over all observations $Y_{[0,T]}$. Optimizing this bound w.r.t. the approximate probability measure yields the exact posterior probability measure $\mathrm{Q}(X_{[0,T]} \in \cdot) = \mathrm{P}(X_{[0,T]} \in \cdot \mid y_{[0,T]}, \phi)$. We note that the bound $\mathrm{L}(\phi)$ can be written as

$$\begin{aligned}
\mathrm{L}(\phi) &= \mathsf{E}_{\mathrm{Q}}\left[\log \frac{\mathrm{d}\,\mathrm{P}}{\mathrm{d}\,\mathrm{Q}}(X_{[0,T]}, y_{[0,T]}, \phi)\right] \\
&= \mathsf{E}_{\mathrm{Q}}\left[\log\left(p(y_{[0,T]} \mid X_{[0,T]}, \phi)\frac{\mathrm{d}\,\mathrm{P}}{\mathrm{d}\,\mathrm{Q}}(X_{[0,T]}, \phi)\right)\right] \\
&= \mathsf{E}_{\mathrm{Q}}\left[\log \frac{\mathrm{d}\,\mathrm{P}}{\mathrm{d}\,\mathrm{Q}}(X_{[0,T]}, \phi)\right] + \sum_{i=1}^{N} \mathsf{E}_{q(x,t_i)}\left[\log p(y_i \mid X, \phi)\right] \\
&= -\mathsf{KL}\big(\mathrm{Q}(X_{[0,T]} \in \cdot) \,\big\|\, \mathrm{P}(X_{[0,T]} \in \cdot \mid \phi)\big) + \sum_{i=1}^{N} \mathsf{E}_{q(x,t_i)}\left[\log p(y_i \mid X, \phi)\right],
\end{aligned}$$

where we denote by $q_t(x) \coloneqq \mathrm{Q}(X(t) = x)$ the time point-wise marginal of the approximate path measure. Note that both the approximate path measure $\mathrm{Q}(X_{[0,T]} \in \cdot)$ and the prior path measure $\mathrm{P}(X_{[0,T]} \in \cdot \mid \phi)$ are path measures induced by an MJP. For the prior path measure $\mathrm{P}(X_{[0,T]} \in \cdot \mid \phi)$ this is easy to note, since $\{X(t)\}$ is an MJP. For the approximate posterior path-measure $\mathrm{Q}(X_{[0,T]} \in \cdot) = \mathrm{P}(X_{[0,T]} \in \cdot \mid y_{[0,T]}, \phi)$ this can be seen, as the equations for its time point-wise marginals $q_t(x) = \mathrm{Q}(X(t) = x) = \mathrm{P}(X(t) = x \mid y_{[0,T]}, \phi) = \tilde{\pi}_t(x)$ are given by the evolution of the smoothing distribution in Eq. (20). This is evolution equation has the form of a master equation with the rate function $\bar{\Lambda}(x, x', t)$ as defined in Eq. (21), and therefore, the posterior process is also an MJP. The KL divergence for two MJPs, which is the expected log Radon-Nikodym derivative, has been derived in the literature multiple times, see, e.g., Kipnis and Landim (1998); Opper and Sanguinetti (2007); Hanson (2007); Cohn et al. (2010). The KL divergence computes to

$$\begin{aligned}
&\mathsf{KL}\big(\mathrm{Q}(X_{[0,T]} \in \cdot) \,\big\|\, \mathrm{P}(X_{[0,T]} \in \cdot \mid \phi)\big) = \mathsf{KL}(q(x,0) \parallel p_0(x \mid \phi)) \\
&+ \int_0^T \mathsf{E}_{q_t(x)}\left[\sum_{x' \neq X} \bar{\Lambda}(X, x', t) \log \frac{\bar{\Lambda}(X, x', t)}{\Lambda(X, x', t \mid \phi)} - \big(\bar{\Lambda}(X, x', t) - \Lambda(X, x', t \mid \phi)\big)\right] \mathrm{d}t,
\end{aligned}$$

where only the prior rates $\Lambda(x, x', t \mid \phi)$ depend on the parameters ϕ. Note, that the forward posterior rates $\bar{\Lambda}(x, x', t)$ in Eq. (21) can be written in terms of the filtering and smoothing distribution by exploiting Eq. (19) as

$$\bar{\Lambda}(x, x', t) = \Lambda(x, x', t)\frac{\beta_t(x')}{\beta_t(x)} = \Lambda(x, x', t)\frac{\tilde{\pi}_t(x')}{\tilde{\pi}_t(x)}\frac{\pi_t(x)}{\pi_t(x')}.$$

Therefore, we arrive at the following expression for the bound as

$$\begin{aligned}
\mathrm{L}(\phi) = \mathsf{E}_{q_0(x)}\left[\log \frac{p_0(X \mid \phi)}{q_0(X)}\right] + \int_0^T \mathsf{E}_{q_t(x)}\Bigg[&\sum_{x' \neq X} \Lambda(X, x', t)\frac{\tilde{\pi}_t(x')}{\tilde{\pi}_t(X)}\frac{\pi_t(X)}{\pi_t(x')} \\
\cdot \log \frac{\Lambda(X, x', t \mid \phi)}{\Lambda(X, x', t)} - \Lambda(X, x', t \mid \phi)&\Bigg] \mathrm{d}t + \sum_{i=1}^{N} \mathsf{E}_{q(x,t_i)}\left[\log p(y_i \mid X, \phi)\right] + \text{const.}
\end{aligned}$$

Since, the computation $\mathrm{Q}(X_{[0,T]} \in \cdot) = \mathrm{P}(X_{[0,T]} \in \cdot \mid y_{[0,T]}, \phi)$ of the exact posterior probability measure is intractable, we replace its time-point wise marginals with the approximate filtering and smoothing distribution as

$p(x,t \mid y_{[0,T]}, \phi) = \tilde{\pi}_t(x) \approx q(x \mid \tilde{\theta}(t))$ and $\pi_t(x) \approx q(x \mid \theta(t))$, respectively. We keep these distributions fixed and this yields an expression for the bound as

$$\begin{aligned}\mathsf{L}(\phi) = \mathsf{E}_{q(x \mid \tilde{\theta}(0)))}\left[\log \frac{p_0(X \mid \phi)}{q(X \mid \tilde{\theta}(0))}\right] + \int_0^T \mathsf{E}_{q(x \mid \tilde{\theta}(t))}\Bigg[\sum_{x' \neq X} \Lambda(X, x', t) \frac{q(x' \mid \tilde{\theta}(t))}{q(X \mid \tilde{\theta}(t))} \frac{q(X \mid \theta(t))}{q(x' \mid \theta(t))} \\ \cdot \log \frac{\Lambda(X, x', t \mid \phi)}{\Lambda(X, x', t)} - \Lambda(X, x', t \mid \phi)\Bigg] \mathrm{d}t + \sum_{i=1}^{N} \mathsf{E}_{q(x \mid \tilde{\theta}(t_i))}\left[\log p(y_i \mid X, \phi)\right] + \text{const.}\end{aligned} \tag{22}$$

Therefore, we can find the optimal parameters, by iteratively computing the filtering and smoothing distribution $\{q(x \mid \theta(t)) \mid t \in [0,T]\}$ and $\{q(x \mid \tilde{\theta}(t)) \mid t \in [0,T]\}$, respectively, and optimizing Eq. (22) w.r.t. the parameters ϕ.

4 DERIVATION FOR CHEMICAL REACTION NETWORKS

4.1 Filtering Distribution

The evolution of the approximate filtering distribution is given by

$$\frac{\mathrm{d}}{\mathrm{d}t}\theta(t) = F(\theta(t))^{-1} \, \mathsf{E}_{q(x \mid \theta(t))}\left[\mathcal{L}_t^{\dagger} \nabla_\theta \log q(X \mid \theta(t))\right].$$

We assume a product Poisson variational distribution $q(x \mid \theta)$ and the adjoint operator $\mathcal{L}_t^{\dagger}$ is characterized by CRN dynamics with mass action kinetics. Therefore, we compute

$$\begin{aligned}&\mathsf{E}_{q(x \mid \theta(t))}\left[\mathcal{L}_t^{\dagger} \nabla_\theta \log q(X \mid \theta(t))\right] \\ &= \mathsf{E}_{q(x \mid \theta(t))}\left[\sum_{x' \neq X} \Lambda(X, x', t) \left(\nabla_\theta \log q(x' \mid \theta(t)) - \nabla_\theta \log q(X \mid \theta(t))\right)\right] \\ &= \mathsf{E}_{q(x \mid \theta(t))}\left[\sum_{x' \neq X} \Lambda(X, x', t) \left(x' - X\right)\right] \\ &= \mathsf{E}_{q(x \mid \theta(t))}\left[\sum_{x' \neq X} \sum_{j=1}^{k} \mathbb{1}(x' = X + \nu_j) \lambda_j(X) \left(x' - X\right)\right] \\ &= \mathsf{E}_{q(x \mid \theta(t))}\left[\sum_{j=1}^{k} \nu_j \lambda_j(X)\right] = \sum_{j=1}^{k} \nu_j \, \mathsf{E}_{q(x \mid \theta(t))}\left[\lambda_j(X)\right] \\ &= \sum_{j=1}^{k} c_j \nu_j \, \mathsf{E}_{q(x \mid \theta(t))}\left[\prod_{i=1}^{n} (X_i)_{\underline{\nu}_{ij}}\right] = \sum_{j=1}^{k} c_j \nu_j \prod_{i=1}^{n} \left(\exp \theta_i(t)\right)^{\underline{\nu}_{ij}}\end{aligned}$$

The last line is computed, by noting that the Poisson random variables are independent and the mth factorial moment of a Poisson random variable with mean λ is given as $\mathsf{E}[(X)_m] = \lambda^m$. By writing this results in the natural parameterization, we arrive at

$$\frac{\mathrm{d}}{\mathrm{d}t}\theta(t) = F(\theta(t))^{-1} \sum_{j=1}^{k} c_j \nu_j \exp\left(\sum_{i=1}^{n} \underline{\nu}_{ij} \theta_i(t)\right).$$

4.2 Kalman-type Updates for the Filtering Distribution

At the observation time points, we have discrete-time resets from $\theta = \theta(t_i^-)$ to $\theta^* = \theta(t)$as

$$\theta^* = \underset{\theta'}{\operatorname{argmin}} \, \mathsf{KL}(p(y_i \mid x) q(x \mid \theta) \parallel q(x \mid \theta')), \quad \forall i = 1, \ldots, N$$

The optimal new parameter $\theta^* = \theta(t)$ can be found by computing the derivative of the KL divergence and setting it to zero. Given the exponential family distribution for $q(x \mid \theta)$ this leads to a moment matching condition as $\mathsf{E}_{q(x|\theta^*)}[s(X)] = \mathsf{E}_{p(x|y_i)}[s(X)]$, where $p(x \mid y_i) = {p(y_i|x)q(x|\theta)}/{\sum_{x'} p(y_i|x')q(x'|\theta)}$ is the exact posterior distribution. Since the Gaussian likelihood $p(y_i \mid x)$ is not conjugate to the product Poisson distribution $q(x \mid \theta)$, the computation of the exact posterior $p(x \mid y_i)$ is generally intractable. However, we can find a conjugate update, by assuming a Gaussian distribution at the observation time points as $q(x \mid \theta) \approx \mathcal{N}(x \mid m, P)$. By first performing a minimization of the KL divergence $\mathsf{KL}(q(x \mid \theta) \parallel \mathcal{N}(x \mid m, P))$, w.r.t. the parameters m and P of the Gaussian distribution, we can find an approximation to the product Poisson distribution $q(x \mid \theta)$. This yields the standard Gaussian approximation of the Poisson distribution, with parameters $m = [\exp \theta_1, \dots, \exp \theta_n]^\top$ and $P = \operatorname{diag}([\exp \theta_1, \dots, \exp \theta_n]^\top)$ This then leads to conjugate updates, as for the new mean parameter $m^* = \mathsf{E}_{p(x|y_i)}[X]$ in the update equations for the Kalman filter, see, e.g., Särkkä (2013), as

$$m^* = m + PH^\top (HPH^\top + \Sigma)^{-1}(y_i - Hm).$$

Finally we can compute the optimal variational parameters after the jump according to the moment matching condition $\mathsf{E}_{q(x|\theta^*)}[s(X)] = \mathsf{E}_{p(x|y_i)}[s(X)]$. Note, that the mean of the Gaussian can be negative, as such we truncate it at a small value $0 < \epsilon_\lambda \ll 1$. This yields

$$\theta^* = \log\left(\max\left\{\exp\theta + F(\theta)H^\top(HF(\theta)H^\top + \Sigma)^{-1}(y_i - H\exp\theta), \epsilon_\lambda\right\}\right).$$

The resulting update therefore can be written as

$$\xi_i = \log\left(\max\left\{\exp\theta + F(\theta)H^\top(HF(\theta)H^\top + \Sigma)^{-1}(y_i - H\exp\theta), \epsilon_\lambda\right\}\right) - \theta.$$

4.3 Smoothing Distribution

For the derivation to the smoothing distribution we compute

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{\theta}(t) = -F(\tilde{\theta}(t))^{-1}\,\mathsf{E}_{q(x|\tilde{\theta}(t))}\left[\tilde{\mathcal{L}}_t^\dagger \nabla_\theta \log q(X \mid \tilde{\theta}(t))\right].$$

The expectation on the r.h.s. computes to

$$\begin{aligned}
&\mathsf{E}_{q(x|\tilde{\theta}(t))}\left[\tilde{\mathcal{L}}_t^\dagger \nabla_\theta \log q(X \mid \tilde{\theta}(t))\right] = \mathsf{E}_{q(x|\tilde{\theta}(t))}\left[\sum_{x' \neq X} \tilde{\Lambda}(X, x', t)\,(x' - X)\right] \\
&= \mathsf{E}_{q(x|\tilde{\theta}(t))}\left[\sum_{x' \neq X} \Lambda(x', X, t)\frac{\pi_t(x')}{\pi(X, t)}\,(x' - X)\right] \\
&= \mathsf{E}_{q(x|\tilde{\theta}(t))}\left[\sum_{x' \neq X} \Lambda(x', X, t)\frac{q(x' \mid \theta(t))}{q(X \mid \theta(t))}\,(x' - X)\right] \\
&= \mathsf{E}_{q(x|\tilde{\theta}(t))}\left[\sum_{x' \neq X}\sum_{j=1}^{k} \mathbb{1}(X = x' + \nu_j)\lambda_j(x')\left\{\prod_{i=1}^{n}\frac{X_i!}{x_i'!}\exp((x_i' - X_i)\theta_i(t))\right\}(x' - X)\right] \\
&= \mathsf{E}_{q(x|\tilde{\theta}(t))}\left[\sum_{j=1}^{k} \lambda_j(X - \nu_j)\left\{\prod_{i=1}^{n}\frac{X_i!}{(X_i - \nu_{ij})!}\exp(-\nu_{ij}\theta_i(t))\right\}(-\nu_j)\right] \\
&= \mathsf{E}_{q(x|\tilde{\theta}(t))}\left[\sum_{j=1}^{k} \lambda_j(X - \nu_j)\left\{\prod_{i=1}^{n}(X_i)_{\nu_{ij}}\right\}\exp\left(-\sum_{i=1}^{n}\nu_{ij}\theta_i(t)\right)(-\nu_j)\right] \\
&= -\sum_{j=1}^{k} c_j\nu_j\exp\left(-\sum_{i=1}^{n}\nu_{ij}\theta_i(t)\right)\mathsf{E}_{q(x|\tilde{\theta}(t))}\left[\left\{\prod_{i=1}^{n}(X_i - \nu_{ij})_{\nu_{ij}}\right\}\left\{\prod_{i=1}^{n}(X_i)_{\nu_{ij}}\right\}\right] \\
&= -\sum_{j=1}^{k} c_j\nu_j\exp\left(-\sum_{i=1}^{n}\nu_{ij}\theta_i(t)\right)\mathsf{E}_{q(x|\tilde{\theta}(t))}\left[\prod_{i=1}^{n}(X_i - \nu_{ij})_{\underline{\nu}_{ij}}(X_i)_{\nu_{ij}}\right]
\end{aligned}$$

$$
\begin{aligned}
&= -\sum_{j=1}^{k} c_j \nu_j \exp\left(-\sum_{i=1}^{n} \nu_{ij}\theta_i(t)\right) \mathsf{E}_{q(x\mid\tilde{\theta}(t))}\left[\prod_{i=1}^{n}(X_i-\nu_{ij})_{\underline{\nu}_{ij}}(X_i)_{\nu_{ij}}\right]\\
&= -\sum_{j=1}^{k} c_j \nu_j \exp\left(-\sum_{i=1}^{n} \nu_{ij}\theta_i(t)\right) \mathsf{E}_{q(x\mid\tilde{\theta}(t))}\left[\prod_{i=1}^{n}(X_i)_{\nu_{ij}+\underline{\nu}_{ij}}\right]\\
&= -\sum_{j=1}^{k} c_j \nu_j \exp\left(-\sum_{i=1}^{n} \nu_{ij}\theta_i(t)\right) \prod_{i=1}^{n}\lambda_i(t)^{\nu_{ij}+\underline{\nu}_{ij}}\\
&= -\sum_{j=1}^{k} c_j \nu_j \exp\left(-\sum_{i=1}^{n} \nu_{ij}\theta_i(t)\right) \exp\left(\sum_{i=1}^{n}(\nu_{ij}+\underline{\nu}_{ij})\tilde{\theta}_i(t)\right)\\
&= -\sum_{j=1}^{k} c_j \nu_j \exp\left(\sum_{i=1}^{n} \underline{\nu}_{ij}\tilde{\theta}_i(t)\right) \exp\left(\sum_{i=1}^{n}\nu_{ij}(\tilde{\theta}_i(t)-\theta_i(t))\right).
\end{aligned}
$$

This yields the smoothing dynamics as

$$
\frac{\mathrm{d}}{\mathrm{d}t}\tilde{\theta}(t) = F(\tilde{\theta}(t))^{-1}\sum_{j=1}^{k} c_j\nu_j \exp\left(\sum_{i=1}^{n}\underline{\nu}_{ij}\tilde{\theta}_i(t)\right)\exp\left(\sum_{i=1}^{n}\nu_{ij}(\tilde{\theta}_i(t)-\theta_i(t))\right).
$$

5 PARAMETER LEARNING FOR LATENT CHEMICAL REACTION NETWORKS

Here, we give an expression for the bound $\mathrm{L}(\phi)$ for the case of a product Poisson approximation for the filtering and smoothing distributions, for dynamics given as a CRN with mass action kinetics, and a linear Gaussian observation model. For the parameters we consider the initial condition parameters $\{\theta_{i,0}\}_{i=0}^{n}$, the effective rate parameters $\{c_j\}_{j=1}^{k}$, and the linear observation model parameter H and observation noise covariance Σ. Therefore, the full set of parameters are $\phi = \{\{\theta_{i,0}\}_{i=0}^{n}, \{c_j\}_{j=1}^{k}, H, \Sigma\}$.

Initial Condition Parameters. First, we note that for the KL divergence

$$
\mathsf{KL}\Big(q(x\mid\tilde{\theta}(0))\,\Big\|\,p_0(x\mid\phi)\Big) = -\,\mathsf{E}_{q(x\mid\tilde{\theta}(0)))}\left[\log\frac{p_0(X\mid\phi)}{q(X\mid\tilde{\theta}(0))}\right]
$$

in Eq. (22) can be computed in closed form, as both distributions are given by a product of Poisson distributions as $q(x\mid\tilde{\theta}(0)) = \prod_{i=1}^{n}\mathrm{Pois}(x_i\mid\exp(\tilde{\theta}_i(0)))$ and $p_0(x\mid\phi)=\prod_{i=1}^{n}\mathrm{Pois}(x_i\mid\exp(\theta_{i,0}))$. Therefore, the KL divergence between two product Poisson distributions computes to

$$
\mathsf{KL}\Big(q(x\mid\tilde{\theta}(0))\,\Big\|\,p_0(x\mid\phi)\Big) = \sum_{i=1}^{n}\exp(\theta_{i,0})-\exp(\tilde{\theta}_i(0))+\exp(\tilde{\theta}_i(0))\log\frac{\exp(\tilde{\theta}_i(0))}{\exp(\theta_{i,0})}.
$$

Hence, for the bound w.r.t. the ith initial condition parameter $\theta_{i,0}$ we have

$$
\mathrm{L}(\theta_{i,0}) = -\exp(\theta_{i,0})+\exp(\tilde{\theta}_i(0))\theta_{i,0}+\mathrm{const.}
$$

Hence, computing the derivative $\frac{\partial L}{\partial\theta_{i,0}}$ and setting it to zero yields the optimal initial parameter as

$$
\theta_{i,0} = \tilde{\theta}_i(0).
$$

Rate Parameters. For optimizing the effective rate parameters $\{c_j\}_{j=1}^{k}$, we have to compute the expectation in Eq. (22) over the MJP rates, which is

$$
\int_0^T \mathsf{E}_{q(x\mid\tilde{\theta}(t))}\left[\sum_{x'\neq X}\Lambda(X,x',t)\frac{q(x'\mid\tilde{\theta}(t))}{q(X\mid\tilde{\theta}(t))}\frac{q(X\mid\theta(t))}{q(x'\mid\theta(t))}\log\frac{\Lambda(X,x',t\mid\phi)}{\Lambda(X,x',t)}-\Lambda(X,x',t\mid\phi)\right]\mathrm{d}t. \tag{23}
$$

First, we note that we can compute the ratios of smoothing and filtering distributions as

$$\frac{q(x' \mid \tilde{\theta}(t))}{q(x \mid \tilde{\theta}(t))} \frac{q(x \mid \theta(t))}{q(x' \mid \theta(t))} = \prod_{i=1}^{n} \frac{\exp((x'_i - x_i)\tilde{\theta}_i(t))}{\exp((x'_i - x_i)\theta_i(t))} = \exp\left(\sum_{i=1}^{n} (x'_i - x_i)(\tilde{\theta}_i(t) - \theta_i(t))\right).$$

For the rate functions $\Lambda(x, x', t)$ and $\Lambda(x, x', t \mid \phi)$ we use the equations for a CRN model finding

$$\Lambda(x, x', t) = \sum_{j=1}^{k} \mathbb{1}(x' = x + \nu_j) c_j^{\text{old}} \prod_{i=1}^{n} (x_i)_{\underline{\nu}_{ij}},$$
$$\Lambda(x, x', t \mid \phi) = \sum_{j=1}^{k} \mathbb{1}(x' = x + \nu_j) c_j \prod_{i=1}^{n} (x_i)_{\underline{\nu}_{ij}},$$

where we denote by $\{c_j^{\text{old}}\}_{j=1}^{k}$ the set of old rate parameters, over which we do not optimize. This yields for the expectation in Eq. (23)

$$\begin{aligned}
&\int_0^T \mathsf{E}_{q(x \mid \tilde{\theta}(t))}\left[\sum_{x' \neq X} \Lambda(X, x', t) \frac{q(x' \mid \tilde{\theta}(t))}{q(X \mid \tilde{\theta}(t))} \frac{q(X \mid \theta(t))}{q(x' \mid \theta(t))} \log \frac{\Lambda(X, x', t \mid \phi)}{\Lambda(X, x', t)} - \Lambda(X, x', t \mid \phi)\right] \mathrm{d}t \\
&= \int_0^T \mathsf{E}_{q(x \mid \tilde{\theta}(t))}\left[\sum_{j=1}^{k} c_j^{\text{old}} \left\{\prod_{i=1}^{n} (X_i)_{\underline{\nu}_{ij}}\right\} \exp\left(\sum_{i=1}^{n} \nu_{ij}(\tilde{\theta}_i(t) - \theta_i(t))\right) \log\left(\frac{c_j}{c_j^{\text{old}}}\right)\right. \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \left. - c_j \prod_{i=1}^{n} (X_i)_{\underline{\nu}_{ij}}\right] \mathrm{d}t \\
&= \sum_{j=1}^{k} \int_0^T \left\{c_j^{\text{old}} \exp\left(\sum_{i=1}^{n} \underline{\nu}_{ij} \tilde{\theta}_i(t)\right) \exp\left(\sum_{i=1}^{n} \nu_{ij}(\tilde{\theta}_i(t) - \theta_i(t))\right) \log\left(\frac{c_j}{c_j^{\text{old}}}\right)\right. \\
&\qquad\qquad\qquad\qquad\qquad\qquad \left. - c_j \exp\left(\sum_{i=1}^{n} \underline{\nu}_{ij} \tilde{\theta}_i(t)\right)\right\} \mathrm{d}t.
\end{aligned}$$

Hence, the bound $\mathrm{L}(c_j)$ w.r.t. the jth rate parameter c_j, can be computed as

$$\begin{aligned}
\mathrm{L}(c_j) = \int_0^T & c_j^{\text{old}} \exp\left(\sum_{i=1}^{n} \underline{\nu}_{ij} \tilde{\theta}_i(t)\right) \exp\left(\sum_{i=1}^{n} \nu_{ij}(\tilde{\theta}_i(t) - \theta_i(t))\right) \log c_j \\
& - c_j \exp\left(\sum_{i=1}^{n} \underline{\nu}_{ij} \tilde{\theta}_i(t)\right) \mathrm{d}t + \text{const.}
\end{aligned}$$

By introducing the summary propensity statistics

$$\hat{\gamma}_j = \frac{1}{T} \int_0^T c_j^{\text{old}} \exp\left(\sum_{i=1}^{n} \underline{\nu}_{ij} \tilde{\theta}_i(t)\right) c_j^{\text{old}} \exp\left(\sum_{i=1}^{n} \nu_{ij}(\tilde{\theta}_i(t) - \theta_i(t))\right) \mathrm{d}t,$$
$$\hat{\lambda}_j = \frac{1}{T} \int_0^T c_j^{\text{old}} \exp\left(\sum_{i=1}^{n} \underline{\nu}_{ij} \tilde{\theta}_i(t)\right) \mathrm{d}t,$$

we write the bound as

$$\mathrm{L}(c_j) = \frac{T}{c_j^{\text{old}}} \hat{\gamma}_j \log c_j - \frac{T}{c_j^{\text{old}}} \hat{\lambda}_j c_j + \text{const.}$$

Hence, computing the derivative $\frac{\partial L}{\partial c_j}$ and setting it to zero yields the optimal rate parameter as

$$c_j = \hat{\gamma}_j \hat{\lambda}_j^{-1}.$$

Observation Model Parameters. The observation model parameters can be found by computing

$$\begin{aligned}\sum_{i=1}^{N} \mathsf{E}_{q(x \mid \tilde{\theta}(t_i))} \left[\log p(y_i \mid X, \phi)\right] &= \sum_{i=1}^{N} \mathsf{E}_{q(x \mid \tilde{\theta}(t_i))} \left[\log \mathcal{N}(y_i \mid HX, \Sigma)\right] \\ &= \sum_{i=1}^{N} -\frac{1}{2} \log |2\pi\Sigma| - \frac{1}{2} \operatorname{tr} \Big\{ \Sigma^{-1} \Big(y_i y_i^{\top} - y_i \, \mathsf{E}_{q(x \mid \tilde{\theta}(t_i))}[X^{\top}] H^{\top} - H \, \mathsf{E}_{q(x \mid \tilde{\theta}(t_i))}[X] y_i^{\top} \\ &\qquad + H \, \mathsf{E}_{q(x \mid \tilde{\theta}(t_i))}[X X^{\top}] H^{\top} \Big) \Big\}.\end{aligned}$$

By using the product Poisson distribution $q(x \mid \tilde{\theta}(t_i)) = \prod_{i=1}^{n} \operatorname{Pois}(x_i \mid \exp(\tilde{\theta}(t_i)))$, we compute the moments as $\mathsf{E}_{q(x \mid \tilde{\theta}(t_i))}[X] = \exp(\tilde{\theta}(t_i))$ and $\mathsf{E}_{q(x \mid \tilde{\theta}(t_i))}[X X^{\top}] = \operatorname{diag}\{\exp(\tilde{\theta}(t_i))\} + \exp(\tilde{\theta}(t_i)) \exp(\tilde{\theta}^{\top}(t_i))$. Therefore, we have the lower bound

$$\mathrm{L}(H, \Sigma) = -\frac{N}{2} \log |2\pi\Sigma| - \frac{N}{2} \operatorname{tr} \Big\{ \Sigma^{-1} \Big(\hat{M}_{YY} - \hat{M}_{XY} H^{\top} - H \hat{M}_{XY}^{\top} + H \hat{M}_{XX} H^{\top} \Big) \Big\} + \text{const},$$

where we introduce the shorthands $\hat{M}_{YY} = \frac{1}{N} \sum_{i=1}^{N} y_i y_i^{\top}$, $\hat{M}_{XY} = \frac{1}{N} \sum_{i=1}^{N} y_i \exp(\tilde{\theta}^{\top}(t_i))$ and $\hat{M}_{XX} = \frac{1}{N} \sum_{i=1}^{N} \operatorname{diag}\{\exp(\tilde{\theta}(t_i))\} + \exp(\tilde{\theta}(t_i)) \exp(\tilde{\theta}^{\top}(t_i))$. Hence, computing the derivatives $\frac{\partial L}{\partial H}$ and $\frac{\partial L}{\partial \Sigma}$, and setting them to zero yield the optimal observation model parameters as

$$H = \hat{M}_{XY} \hat{M}_{XX}^{-1}, \quad \Sigma = \hat{M}_{YY} - \hat{M}_{XY} H^{\top} - H \hat{M}_{XY}^{\top} + H \hat{M}_{XX} H^{\top}.$$

Note that, this is very similar to the case of a linear Gaussian state space model, for more see Särkkä (2013).

6 ADDITIONAL EXPERIMENTAL RESULTS

In this section, we provide further details on the experiments presented in the main document, along with additional experiments.

6.1 Benchmark Tasks

6.1.1 Lotka-Volterra Model

For the Lotka-Volterra task discussed in the experiments section of the main paper, we evaluate our method and the baseline methods on 100 sample trajectories with time horizon $T = 300$. Each latent trajectory gets observed at 10 randomly chosen, non-equidistant time points with linear Gaussian measurements of the latent state, i.e., we assume the observation likelihood $p(y_i \mid x) = \mathcal{N}(y_i \mid Hx, \Sigma)$. The learning rate for the expectation propragation algorithm, also called dampening parameter is set to $\epsilon = 0.05$ for this and all other experiments. We summarize the parameters in Table 2. The results of the experiments are highlighted in the main paper.

Table 2: Parameters of the Lotka-Volterra experiments

Parameter	**Value**
rate c_1	0.005
rate c_2	0.001
rate c_3	0.005
observation matrix H	$\begin{bmatrix}1 & 0 \\ 0 & 1\end{bmatrix}$
observation covariance matrix Σ	$\begin{bmatrix}1 & 0 \\ 0 & 1\end{bmatrix}$

6.1.2 Motility Model

For our motility model experiment, we consider a sample trajectory with time horizon $T = 200$. The latent trajectory is observed at 5 randomly chosen, non-equidistant time points with linear Gaussian measurements

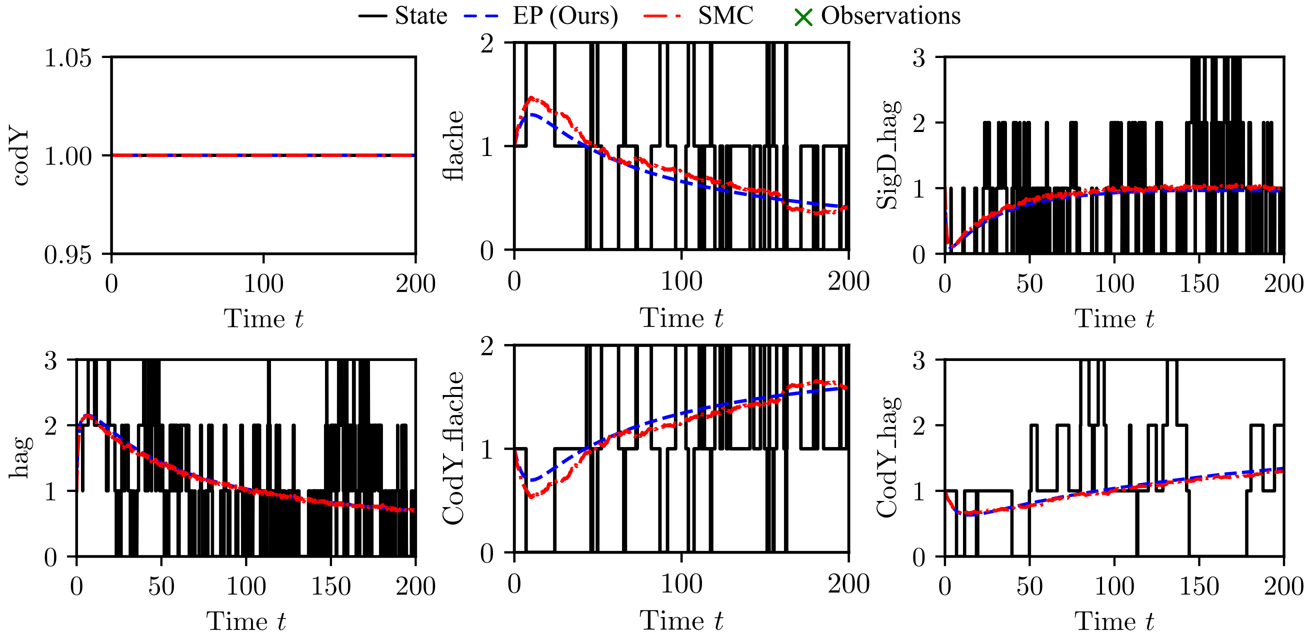

Figure 4: Additional results of the motility model. Here, the posterior mean tracks roughly the time-average of the ground-truth signal. However, since for this species the effective realizations are very low, random fluctuations play a more significant role, making accurate tracking more challenging. The results of our method closely aligns with the SMC results.

of the SigD species. The parameters of the experiment are summarized in Table 3. The additional results of the motility model experiment are depicted in Fig. 4. This experiment highlights the particle degeneracy issue of the SMC method, which we describe shortly. In our implementation of the SMC method, we start with $N_s = 10000$ uniformly weighted samples. At each observation point, we update the weights based on the observation likelihood and then resample the trajectories according to the updated weights. This process leads to repeated resampling of the early section of the trajectories, resulting in a significant reduction in the number of distinct trajectories in these sections. In our experiment, the section until the first observation time point of the latent estimate is described by only 1734 distinct trajectories out of the 10000 samples. While this still provides a good approximation of the ground truth, the number of distinct trajectories decreases even further for longer or more complex tasks, necessitating more samples to maintain accuracy. Although SMC with a large number of samples serves as a reliable approximation of the ground truth for comparison in our study, the number of samples required increases significantly with model complexity and time horizon, making SMC less scalable and less practical for higher-dimensional or longer-horizon problems. In contrast, the computational cost of our method scales linearly with the time horizon, as it primarily depends on solving a system of ODEs, where the number of required computations increases proportionally with the length of the time horizon.

6.1.3 Gene Transcription and Translation

As an additional benchmark we consider a stochastic model representing the gene transcription and translation (Anderson and Kurtz, 2015). Transcription refers to the process of copying information encoded in the DNA to a messenger RNA (mRNA). The translation of an mRNA by a ribosome yields proteins.

The model is defined by the following reactions:

$$\mathrm{G} \xrightarrow{c_1} \mathrm{G} + \mathrm{M}, \quad \mathrm{M} \xrightarrow{c_2} \mathrm{M} + \mathrm{P}, \qquad \mathrm{M} \xrightarrow{c_3} \emptyset, \qquad \mathrm{P} \xrightarrow{c_4} \emptyset,$$

where G represents gene, M the mRNA and P the proteins. The reactions represent the transcription, translation, degradation of mRNA and the degradation of proteins, respectively.

Similar to the Lotka-Volterra experiment, we infer the approximate posterior for 100 sample trajectories with time horizon $T = 8$. Each latent trajectory is observed at 10 randomly chosen, non-equidistant time points with linear

Table 3: Parameters of the motility experiment

Parameter	**Value**
rate c_1	0.1
rate c_2	0.0002
rate c_3	1.
rate c_4	0.0002
rate c_5	1.0
rate c_6	0.0002
rate c_7	0.01
rate c_8	0.1
rate c_9	0.02
rate c_{10}	0.1
rate c_{11}	0.01
rate c_{12}	0.1
observation covariance σ^2	100

Gaussian measurements of the protein species. We summarize the parameters in Table 4. We again compare the results to the baseline results by (i) a single FFBS iteration using the entropic matching method without EP, (ii) a Gaussian ADS based on the chemical Langevin equation, similar to the method described by Cseke et al. (2016), (iii) the moment-based VI method proposed by Wildner and Koeppl (2019), and (iv) an exact smoothing algorithm based on a truncation of the system, which serves as ground truth for our comparison.

Table 4: Parameters of the gene transcription and translation experiment

Parameter	**Value**
rate c_1	200
rate c_2	10
rate c_3	25
rate c_4	1
observation covariance σ^2	10

Table 5: Mean squared error in posterior mean averaged over trajectories and time for the gene transcription and translation experiment

EP (Ours)	FFBS Entropic	G ADS	MBVI
0.1919	2.9912	652.68	2.2687

Table 5 shows the mean squared error in the posterior mean of the approximate methods compared to the exact posterior mean of the truncated system. Similar to the Lotka-Volterra experiment, our method demonstrates superior performance overall. Notably, the Gaussian ADS performs poorly, which we attribute to its inappropriate modeling choice of representing low population counts with Gaussian distributions. We visualize the result of one sample trajectory in Fig. 5.

6.1.4 Enzyme Kinetics

Finally, as last benchmark we study the enzyme kinetics model, a standard model in which an enzyme catalyzes the conversion of some substrate to product (Anderson and Kurtz, 2015).

The model is defined by the following reactions:

$$\mathrm{S} + \mathrm{E} \xrightarrow{c_1} \mathrm{SE}, \quad \mathrm{SE} \xrightarrow{c_2} \mathrm{S} + \mathrm{E}, \qquad \mathrm{SE} \xrightarrow{c_3} \mathrm{P} + \mathrm{E},$$

where S is the substrate species, E the enzyme, SE an enzyme-substrate and P the product species.

Again, we infer the approximate posterior for 100 sample trajectories with time horizon $T = 20$. Each latent trajectory is observed at 10 randomly chosen, non-equidistant time points with linear Gaussian measurements of

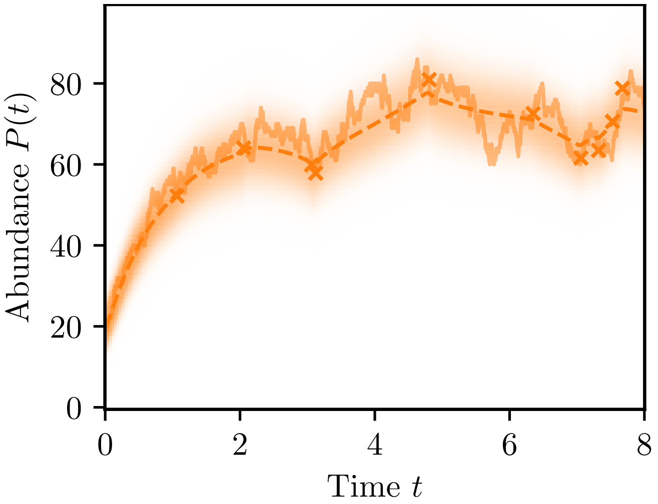

Figure 5: Simulation of the Gene Expression model. The protein species is visualized in orange. Dashed lines denote variational posterior mean, solid lines denote ground truth trajectory, the background indicates the inferred marginal state probabilities and the crosses indicate observations.

the protein and substrate species. We summarize the parameters in Table 6. We again compare the results to the previously mentioned baselines.

Table 6: Parameters of the enzyme kinetics experiment

Parameter	**Value**
rate c_1	0.05
rate c_2	0.5
rate c_3	0.5
observation covariance matrix Σ	$\begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$

Table 7: Mean squared error in posterior mean averaged over trajectories and time for the gene transcription and translation experiment

EP (Ours)	FFBS Entropic	G ADS	MBVI
0.3339	1.3432	7.0121	0.6091

Table 7 shows the mean squared error in the posterior mean of the approximate methods compared to the exact posterior mean of the truncated system. Like in the previous cases, out method demonstrates superior performance in estimating the posterior mean. We visualize the result of one sample trajectory in Fig. 6.

6.2 Parameter Learning

In this section we analyze the task of inferring the latent state when not all parameters of the model are known. To address this, we employ the previously described approximate EM algorithm to jointly infer both the parameters and the latent state. We focus on experiments with unknown rate parameters. Closed form solutions for the M-Step of the EM algorithm are derived in Appendix 5. Notably, in our experiments we find that using the entropic matching method with a single FFBS iteration yields parameter estimation results comparable to those using EP method with entropic matching. This enables a significantly faster algorithm, as only one FFBS iteration needs to be computed per EM step. Finally for the latent state estimation we use the proposed EP algorithm with the resulting parameters of the final EM step.

We apply the EM algorithm to the Lotka-Volterra model, the Gene transcription and translation task and the enzyme kinetics model, using the same data as in the previous experiments. Specifically, for each task, we run

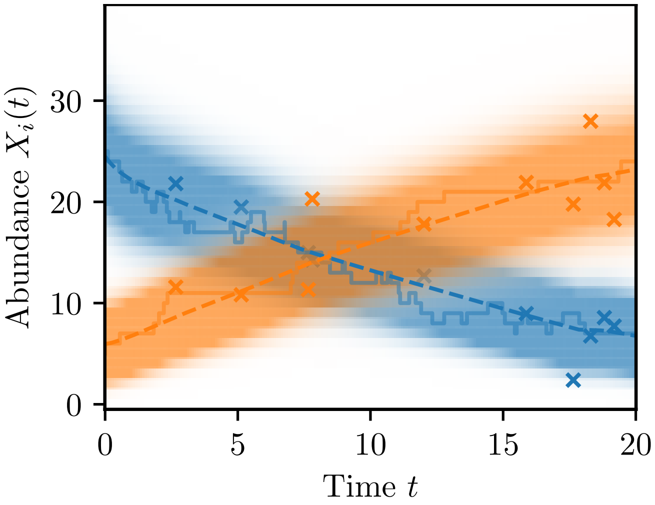

Figure 6: Simulation of the Enzyme Kinetics model. The abundance of the substrate species is visualized in blue, the product species in orange. Dashed lines denote variational posterior mean, solid lines denote ground truth trajectory, the background indicates the inferred marginal state probabilities and the crosses indicate observations.

the EM algorithm on each of the 100 sample trajectories with the same observations as described earlier. For the Lotka-Volterra model, we assume all rate parameters to be unknown; for the gene translation and transcription task we assume c_2 to be unknown; and for the enzyme kinetics task we treat c_3 as unknown.

We initialize all parameters with starting guesses in the same order of magnitude as their true values, ensuring a realistic but non-trivial starting point for the inference process. Table 8 shows the true parameter values, the initial guesses for the EM algorithm and the estimated values, averaged across the 100 sample trajectories. We observe that our proposed method yields reasonably accurate results for the Lotka-Volterra task, where all rate parameters are unknown and very good results for the other tasks where only one parameter is unknown.

Table 8: Estimated Parameter Values

Parameter	**True Value**	**Initial Value**	**Estimated Value**
LV c_1	0.005	0.002	0.0077
LV c_2	0.001	0.002	0.0012
LV c_3	0.005	0.002	0.0064
Gene c_2	10.0	5.0	9.963
Enzyme c_3	0.5	0.1	0.520

Further very interesting are the results for the latent state inference. We compute the approximate posterior mean for all tasks and compare it to the results of the exact method based on truncation, assuming full knowledge of the parameters. The error averaged over time and trajectories is summarized in Table 9. When comparing these results with Tables 1, 5 and 7, we find that our proposed method for joint inference of parameters and latent state demonstrates superior performance in estimating the mean compared to the baselines methods that utilize full knowledge of the parameters. Naturally, the EP algorithm with full knowledge of the parameters performs even better. This underscores the efficacy of the EM algorithm.

Table 9: Mean squared error in posterior mean averaged over trajectories and time for all tasks

LV	Gene	Enzyme
0.7955	1.0936	0.4289

6.3 Computational Cost

The computational cost of our proposed method is primarily driven by the repeated solution of the ODEs in Eqs. (16) and (18). Since we derived closed-form solutions for the right-hand sides of these equations, we can evaluate these ODEs efficiently and fast. For the numerical integration, we utilized the Runge-Kutta-Fehlberg method from the SciPy package. All computations were performed on an Apple M1 chip.

Similarly, the computational cost of the moment-based VI method by Wildner and Koeppl (2019) is dominated by repeatedly solving ODEs for a forward and a backward pass. However, this method approximates the first and second-order moments of the latent state distribution, which increases the dimensionality of the ODE system significantly, making this approach less scalable for high-dimensional problems.

The Gaussian ADS approximates first and second-order moments similar to the VI method, and therefore encounters the same scalability issues as the number of species increases. However the Gaussian ADS is significantly faster, as it requires only one forward and backward pass.

In contrast, the computational cost of the SMC method depends on the number of samples and the cost of generating a sample trajectory. As discussed earlier, the particle degeneracy issue can significantly increase the number of particles required to maintain an accurate approximation, making this approach less scalable, particularly for longer time horizons or high-dimensional problems.

6.4 Discussion

The experimental results demonstrate the effectiveness and scalability of our proposed method for latent state inference and parameter learning in complex models. However, it is important to acknowledge that other methods have their own merits, and the choice of approach depends largely on the specific use case and model characteristics.

For latent state inference in MJPs with possibly unknown parameters and low to moderate population sizes, we recommend our proposed method. In contrast, for systems with large population sizes where the underlying MJP can be well-approximated by an SDE, the Gaussian ADS offers a suitable alternative.

When the focus extends beyond posterior marginals to the full posterior path, the moment-based variational inference method proposed by Wildner and Koeppl (2019) is a strong option. Alternatively, the mean-field approach proposed by Opper and Sanguinetti (2007) can be employed, which however, is limited to models where only one species changes per jump. In systems where multiple species change simultaneously (as in our LV model), this assumption breaks down, leading to issues with absolute continuity, as discussed by Wildner and Koeppl (2019). The neural variational inference method by Seifner and Sánchez (2023) either integrates the chemical master equation, which is only tractable for small systems or they use the mean-field approach by Opper and Sanguinetti (2007).

SMC methods (Doucet et al., 2001), while computationally intensive and subject to particle degeneracy over long time horizons, can still provide reliable results when configured with a sufficiently large number of particles. They remain a useful choice, particularly when high accuracy is required for state estimation over shorter sequences. Finally, if parameter learning is the primary goal, without the need for detailed latent state inference, MCMC methods provide a Bayesian approach that offers posterior distributions over parameters instead of point estimates (Golightly and Wilkinson, 2011; Golightly et al., 2015; Lowe et al., 2023).