

Broadly Applicable Approximate MCMC for Switching Stochastic Differential Equations Using Uniformization and Time-Conditioned Factorized Neural Likelihood Estimation

Shion Hosoda

shion.hosoda@gmail.com

Waseda University <https://orcid.org/0000-0001-5093-6833>

Michiaki Hamada

Waseda University <https://orcid.org/0000-0001-9466-1034>

Research Article

Keywords: Markov chain Monte Carlo, simulation-based inference, stochastic differential equation

Posted Date: October 6th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11258090/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: The authors declare no competing interests.

Broadly Applicable Approximate MCMC for Switching Stochastic Differential Equations Using Uniformization and Time-Conditioned Factorized Neural Likelihood Estimation

Shion Hosoda^{1,*}

Michiaki Hamada^{1,2,3}

¹ Waseda University

² National Institute of Advanced Industrial Science and Technology

³ Nippon Medical School

Abstract

Switching stochastic differential equations (SSDEs) describe continuous-time dynamics whose parameters switch according to a latent regime process that follows a continuous-time Markov chain (CTMC). By allowing dynamics to change between regimes, SSDEs represent heterogeneous system behavior and have been applied across diverse fields. However, Bayesian inference for SSDEs remains difficult, and existing SSDE inference methods have limited applicability, with restrictions such as noise-free observations, univariate states, linear drift, or state-independent diffusion. In this study, we propose an approximate Markov chain Monte Carlo sampler for SSDEs using uniformization and factorized neural likelihood estimation (FNLE), a simulation-based inference method. Uniformization provides an exact representation of the CTMC but requires SDE transition densities over arbitrary time intervals. We approximate these densities by training a time-conditioned FNLE model. The resulting sampler is broadly applicable to SSDEs without requiring analytically tractable transition densities. In synthetic-data experiments, our method recovered regime paths and parameters for three SSDE models for which previous methods have limited applicability. We also applied our method to a real dataset and detected a regime transition.

1 INTRODUCTION

Stochastic differential equations (SDEs) provide a natural framework for modeling noisy continuous-time dynamics. A single SDE may be insufficient in many cases because dynamics in real-world systems often change. Switching stochastic differential equations (SSDEs) represent such dynamics by allowing the drift and diffusion functions to switch according to the regime at each time. SSDEs have been applied in many fields, such as ecology [36], biology [15], and economics [22]. Figure 1 illustrates an SSDE and its noisy observations.

Bayesian inference for SSDEs remains challenging. It requires joint estimation of latent states, regimes, and model parameters from noisy observations. Moreover, regime transitions can occur between observation times. Due to these challenges, existing SSDE inference methods have limited applicability and are typically restricted to settings with noise-free observations, univariate states, linear drift, or state-independent diffusion [13, 19, 20, 32].

Uniformization is a method to represent the continuous-time Markov chain (CTMC) by a discrete-time regime sequence on Poisson event times [30]. When regime-dependent emission likelihoods are available, this representation facilitates parameter inference in CTMC-based models and has been used in a range of applications [9, 12, 28]. For an SSDE, however, the required likelihood involves transition densities of the state process over each interval. These densities are generally unavailable for SDEs, limiting the direct use of uniformization-based regime updates in general SSDEs.

Simulation-based inference can learn otherwise intractable densities from simulations [5, 29]. Neural likelihood estimation learns conditional densities from simulated parameter–data pairs, while factorized neu-

*Corresponding author: shion.hosoda@gmail.com.

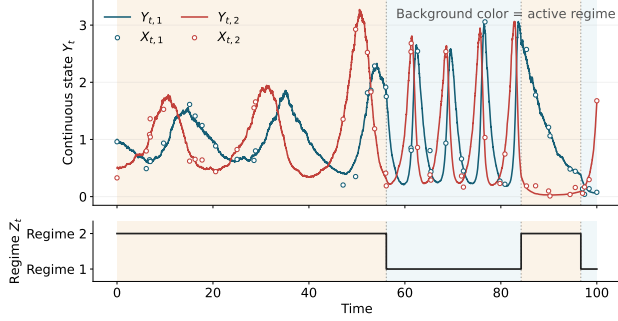


Figure 1: Illustration of a simulated Lotka–Volterra SSDE. The upper panel shows the two components of the continuous state and their noisy observations. Background shading indicates the active regime, and the lower panel shows the regime path.

ral likelihood estimation (FNLE) exploits the Markov property to approximate trajectory likelihoods from learned local transition densities [11]. Standard FNLE is formulated for discrete-time Markovian sequences and is typically trained for a fixed transition interval, and it cannot be directly evaluated at the varying intervals introduced by uniformization.

In this study, we propose an approximate Markov chain Monte Carlo (MCMC) sampler for SSDEs using time-conditioned FNLE. As a foundation, we develop an exact uniformization-based sampler for SSDEs with tractable transition densities and conditional intermediate-state sampling. We then condition FNLE on elapsed time to approximate regime-specific transition densities over varying intervals and use sampling-importance-resampling (SIR) to approximate conditional intermediate-state sampling. This yields a broadly applicable sampler for SSDEs without requiring analytically tractable transition densities.

Our contributions are summarized as follows:

- We combine time-conditioned FNLE with uniformization to construct an approximate MCMC sampler applicable to a broad range of SSDEs.
- We show that the sampler successfully recovers regime paths and model parameters in synthetic experiments on three SSDE models spanning nonlinear drift, state-dependent diffusion, and multivariate latent states.
- We apply the sampler to a real dataset and infer a persistent regime change.

1.1 Related Work

Existing continuous-time SSDE inference methods differ in their model restrictions and treatment of uncertainty. Blackwell [2] considers SSDEs with regimes observed at discrete times. Blackwell et al. [3] use uniformization for SDEs with state-dependent switching and tractable transition densities, jointly updating candidate times, states, and regimes using Metropolis–Hastings (MH). In contrast, our sampler updates regimes conditional on the grid states using a Gibbs update. Hibbah et al. [13] develop a data-augmentation MCMC method for nonlinear SDEs with state-dependent switching, but demonstrate only univariate models without observation error and note computational and mixing difficulties under dense augmentation. Köhs et al. [19, 20] demonstrate multivariate inference with observation error under more restrictive models: the former uses variational inference with state-independent diffusion, while the latter uses MCMC with linear-Gaussian dynamics. Stumpf–Fétizon et al. [32] provide discretization-free exact inference for nonlinear univariate SSDEs with state-dependent diffusion and direct observations, while multivariate extensions require substantially stronger conditions and are not demonstrated.

Our approach combines uniformization-based CTMC sampling [30] with FNLE, a simulation-based inference method for Markovian time series [11]. We use time-conditioned FNLE to approximate SDE transition densities over varying time intervals. Similarly, neural stochastic flows learn time-conditioned SDE transition distributions [17] but are formulated without conditioning on SDE parameters. In contrast, our transition-density approximation is conditioned on both time intervals and model parameters, allowing it to be incorporated into an approximate MCMC sampler that jointly infers latent states, regimes, and model parameters. Table 1 summarizes the differences between our method and existing SSDE inference approaches.

2 BACKGROUND

2.1 Switching Stochastic Differential Equations

A switching stochastic differential equation (SSDE) combines a continuous latent state with a regime that determines its drift and diffusion functions [27]. Let $Y_t \in \mathbb{R}^D$ be a continuous latent state and let $W_t \in \mathbb{R}^D$ be a standard Brownian motion. The regime $Z_t \in \{1, \dots, K\}$ evolves as a CTMC, where D is the state dimension and K is the number of regimes. We con-

Table 1: Applicability of SSDE Inference Methods

Method	Observation error	Nonlinear drift	State-dependent diffusion	Intractable transition density	Multivariate states
Blackwell et al. [3]	$\triangle$	$\triangle$	$\triangle$	$\times$	$\checkmark$
Hibbah et al. [13]	$\triangle$	$\checkmark$	$\checkmark$	$\checkmark$	$\triangle$
Köhs et al. [19]	$\checkmark$	$\triangle$	$\times$	$\triangle$	$\checkmark$
Köhs et al. [20]	$\checkmark$	$\times$	$\times$	$\times$	$\checkmark$
Stumpf-Fétizon et al. [32]	$\triangle$	$\checkmark$	$\checkmark$	$\checkmark$	$\triangle$
Proposed method	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$

$\checkmark$: Demonstrated applicability; $\triangle$: Applicability not established; $\times$: Excluded by model assumptions.

sider an SSDE of the form

$$dY_t = f(Y_t, Z_t)dt + g(Y_t, Z_t)dW_t, \quad (1)$$

where $f : \mathbb{R}^D \times \{1, \dots, K\} \rightarrow \mathbb{R}^D$ and $g : \mathbb{R}^D \times \{1, \dots, K\} \rightarrow \mathbb{R}^{D \times D}$ are the drift and diffusion functions, respectively. For each regime k , $f(\cdot, k)$ and $g(\cdot, k)$ define the corresponding regime-specific SDE. The transition rate matrix $Q = (q_{kh})$ of Z_t satisfies

$$q_{kh} \geq 0 \quad (k \neq h), \quad q_{kk} = -\sum_{h \neq k} q_{kh}. \quad (2)$$

On the observation interval $[0, T]$, we write $Y = \{Y_t\}_{t \in [0, T]}$ and $Z = \{Z_t\}_{t \in [0, T]}$ for the state and regime paths. Their realizations are denoted by $y = \{y_t\}_{t \in [0, T]}$ and $z = \{z_t\}_{t \in [0, T]}$.

2.2 Uniformization of Continuous-Time Markov Chains

Uniformization represents a CTMC as a discrete-time Markov chain observed at event times of a Poisson process [30]. For a fixed observation grid, augmenting the regime path Z with candidate times $\mathcal{T}_{\text{cand}}$ gives an equivalent discrete representation through the almost-sure one-to-one mapping

$$(\tilde{Z}_{1:L}, \mathcal{T}_{\text{event}}) = \Psi(Z, \mathcal{T}_{\text{cand}}), \quad (3)$$

where $\mathcal{T}_{\text{event}} = \mathcal{T}_{\text{true}} \cup \mathcal{T}_{\text{cand}}$ combines the true jump times $\mathcal{T}_{\text{true}}$, at which the regime changes, and candidate times $\mathcal{T}_{\text{cand}}$. For N observations at fixed times $\mathcal{T}_{\text{obs}} = \{t_i^{\text{obs}}\}_{i=1}^N$, with $0 = t_1^{\text{obs}} < \dots < t_N^{\text{obs}} = T$, the combined grid is

$$\mathcal{T}_{\text{all}} = \mathcal{T}_{\text{obs}} \cup \mathcal{T}_{\text{event}} = \{t_1, \dots, t_L\}, \quad 0 = t_1 < \dots < t_L = T. \quad (4)$$

Here, $\mathcal{T}_{\text{event}}$ and $\mathcal{T}_{\text{obs}}$ are almost surely disjoint, and $\tilde{Z}_j = Z_{t_j}$. The sequence $z_{1:L} = (z_{t_1}, \dots, z_{t_L})$ is its realization, distinct from the full path z . Under the right-continuous convention, z_j is the regime on $[t_j, t_{j+1})$ for $j = 1, \dots, L-1$; z_1 and z_L are the initial and terminal regimes.

As an illustrative model, consider conditionally independent latent states with densities $p(\tilde{y}_j | z_j, \theta)$ and conditionally independent observations with densities $p(x_i | \tilde{y}_{\ell(i)}, \theta)$, where θ is fixed and $t_{\ell(i)} = t_i^{\text{obs}}$. Given $\tilde{y}_{1:L}$ and θ , observation factors cancel from the regime conditional. Writing $e_j(k) = p(\tilde{y}_j | z_j = k, \theta)$ and $\mathcal{E} = \{e_j(k)\}_{j=1, \dots, L; k=1, \dots, K}$, we obtain

$$p(z_{1:L} | \mathcal{T}_{\text{all}}, \mathcal{E}, Q) \propto p(z_1) e_1(z_1) \prod_{j=2}^L A_j(z_{j-1}, z_j) e_j(z_j). \quad (5)$$

The transition matrix A_j depends on Q , the uniformization rate Ω , and whether t_j is an event or observation-only time (Appendix A.1). This hidden Markov model (HMM) factorization permits forward-filtering backward-sampling (FFBS) [8].

Conditional on the current path and Q , candidate times are sampled as virtual-jump times from a Poisson process with rate $\Omega(Q) + q_{Z_t Z_t}$, where $\Omega(Q) > \max_k (-q_{kk})$ is deterministic given Q (Appendix A.1). After FFBS, candidate times may become true jump times, and we update

$$\mathcal{T}_{\text{true}} \leftarrow \{t_j \in \mathcal{T}_{\text{all}} : z_{j-1} \neq z_j, j = 2, \dots, L\}. \quad (6)$$

2.3 Factorized Neural Likelihood Estimation for Transition Densities

Factorized neural likelihood estimation (FNLE) is a method for approximating intractable transition densities in Markov process models [11].

For a discrete-time Markov trajectory $y_{1:J}$ of length J , the trajectory likelihood factorizes as

$$p(y_{1:J} | \theta) = p(y_1) \prod_{j=1}^{J-1} p(y_{j+1} | y_j, \theta), \quad (7)$$

where θ denotes the parameters that determine the transition density.

FNLE learns $\hat{p}_\phi(y_{j+1} | y_j, \theta) \approx p(y_{j+1} | y_j, \theta)$ by maximizing the conditional log-likelihood of simulated

transitions, where ϕ denotes the density-estimator parameters. Conditional normalizing flows support both density evaluation and sampling [11, 29], enabling likelihood approximation and SIR proposals in our sampler.

3 PROBLEM SETTING

We define the model targeted by our proposed sampler. We assume that the drift and diffusion functions share common functional forms across regimes as follows:

$$f(y, k) = f(y; \theta_k), \quad g(y, k) = g(y; \theta_k), \quad (8)$$

where θ_k is the SDE parameter vector in regime k . Note that our method is not limited to this case and that functional forms can depend on regimes. We let Θ collect the unconstrained SDE parameters, with separate entries for regime-specific parameters and a single entry for each shared parameter, and write

$$\theta_k = h_k(\Theta), \quad k = 1, \dots, K. \quad (9)$$

Here, h_k selects the parameters used in regime k and applies the transformations required by their constraints.

At the discrete observation times in $\mathcal{T}_{\text{obs}}$, observations are modeled as

$$X_i \mid Y_{t_i^{\text{obs}}}, \tau^2 \sim \mathcal{N}(Y_{t_i^{\text{obs}}}, \text{diag}(\tau^2)). \quad (10)$$

Here, $X_i \in \mathbb{R}^D$ denotes the random observation at time t_i^{obs} , and x_i denotes its realized value. We write $X = \{X_i\}_{i=1}^N$ and $x = \{x_i\}_{i=1}^N$. Furthermore, $\tau^2 \in \mathbb{R}_+^D$ is the vector of observation-noise variances. The initial state and regime have independent priors $p(y_1)$ and $p(z_1)$ that do not depend on the unknown parameters. Prior settings for unknown variables are provided in Appendix D.1.2. The target posterior distribution is written as

$$p(y, z, \Theta, \tau^2, Q \mid x) \propto p(x, y, z, \Theta, \tau^2, Q) \quad (11)$$

$$= p(x \mid y, \tau^2) p(y \mid z, \Theta) p(z \mid Q) \times p(\Theta) p(\tau^2) p(Q). \quad (12)$$

4 PROPOSED METHOD

We first construct an exact MCMC sampler for tractable SSDEs, then use time-conditioned FNLE to obtain an approximate sampler for intractable SSDEs. We call an SSDE tractable when exact transition-density evaluation and exact sampling from the corresponding bridge distribution are available for each regime-specific SDE. We refer to all other SSDEs as intractable in this sense. Here, a bridge distribution is

the conditional distribution of the state path between two fixed times, given the endpoint states and the SDE parameters. We consider a single series; Appendix C.3 gives the extension to multiple series. On the combined grid in Eq. (4), write $y_j = y_{t_j}$ for $j = 1, \dots, L$ and $\Delta_j = t_j - t_{j-1}$ for $j = 2, \dots, L$. We write $Y_{\mathcal{T}_{\text{all}}}$ and $Y_{\mathcal{T}_{\text{all}}}$ for the restrictions of Y to the combined grid and the remaining times, respectively.

4.1 Exact MCMC for Tractable SSDEs

Uniformization augments the posterior in Eq. (12) with $\mathcal{T}_{\text{cand}}$. For fixed $\mathcal{T}_{\text{event}}$, consider successive block Gibbs updates of $Y = (Y_{\mathcal{T}_{\text{all}}}, Y_{\mathcal{T}_{\text{all}}})$, $(Z, Y_{\mathcal{T}_{\text{all}}})$, and $(\Theta, Y_{\mathcal{T}_{\text{all}}})$. The unused draws of $Y_{\mathcal{T}_{\text{all}}}$ can be trimmed by partially collapsed Gibbs sampling [7], leaving the following finite-dimensional updates. Consequently, only the grid values $y_{1:L}$ are explicitly sampled; the intervening paths are marginalized.

Candidate Times Given the current regime path and Q , we sample $\mathcal{T}_{\text{cand}}$ using the Poisson–Uniform construction in Section 2.2 and set $\mathcal{T}_{\text{event}}$ and $\mathcal{T}_{\text{all}}$. The rate Ω and uniformized transition matrix B are computed as in Appendix A.1.

Latent States at Newly Inserted Candidate Times Before updating all grid values, we sample each block $y_{a:b}$ of state values at candidate times from the bridge distribution

$$p(y_{a:b} \mid y_{a-1}, y_{b+1}, z_{1:L}, \Delta_{a:b+1}, \Theta), \quad (13)$$

where $y_{a:b}$ is a consecutive block of $\{y_t\}_{t \in \mathcal{T}_{\text{cand}}}$, with fixed neighboring values y_{a-1} and y_{b+1} at adjacent points of $\mathcal{T}_{\text{obs}} \cup \mathcal{T}_{\text{true}}$.

For SSDEs, we redefine e_j as the exact transition density for $j = 2, \dots, L$:

$$e_j(k) = p(y_j \mid y_{j-1}, \theta_k, \Delta_j). \quad (14)$$

Because z_{j-1} is the regime on $[t_{j-1}, t_j)$, the transition to y_j contributes $e_j(z_{j-1})$.

Latent States on the Combined Time Grid We update $y_{1:L}$ using an MH kernel targeting

$$\begin{aligned} & p(y_{1:L} \mid z_{1:L}, \mathcal{T}_{\text{all}}, \Theta, \tau^2, x) \\ & \propto p(y_1) \prod_{j=2}^L e_j(z_{j-1}) \\ & \times \prod_{i=1}^N p(x_i \mid y_{t_i^{\text{obs}}}, \tau^2). \end{aligned} \quad (15)$$

Regime Path Marginalizing $Y_{\setminus \mathcal{T}_{\text{all}}}$ gives

$$p(z_{1:L} \mid y_{1:L}, \mathcal{T}_{\text{event}}, \Theta, Q, \tau^2, x) \propto p(z_1) \prod_{j=2}^L A_j(z_{j-1}, z_j) e_j(z_{j-1}). \quad (16)$$

This HMM factorization permits FFBS; Appendix A.2.3 gives the derivation.

SDE Parameters The conditional density is

$$p(\Theta \mid y_{1:L}, z_{1:L}, \mathcal{T}_{\text{all}}) \propto p(\Theta) \prod_{j=2}^L e_j(z_{j-1}). \quad (17)$$

We update Θ using an MH kernel targeting this conditional.

Observation Noise and Transition Rates We sample τ^2 and the off-diagonal entries of Q from their conjugate full conditionals, with virtual jumps marginalized in the Q update; see Appendix C.2.

With exact sampling from the bridge distribution and conditional-invariant MH kernels, this sweep preserves the exact posterior marginal of the regime path, latent values at $\mathcal{T}_{\text{obs}} \cup \mathcal{T}_{\text{true}}$, and parameters [25]. Appendix A.2 gives the full derivation.

4.2 Time-Conditioned FNLE for SDE Transition Densities

To construct the approximate MCMC sampler, we approximate the SDE transition densities required by the exact updates. For a regime-specific SDE, the transition density is

$$p(y^{\text{next}} \mid y^{\text{prev}}, \theta_k, \Delta). \quad (18)$$

Here, y^{prev} and y^{next} are state values separated by elapsed time Δ . We train an FNLE model with the elapsed time Δ as a conditioning variable to obtain

$$\hat{p}_\phi(y^{\text{next}} \mid y^{\text{prev}}, \theta_k, \Delta) \approx p(y^{\text{next}} \mid y^{\text{prev}}, \theta_k, \Delta). \quad (19)$$

A single FNLE model can be shared across regimes with common drift and diffusion forms by conditioning on θ_k ; otherwise, a separate model can be trained for each distinct pair of functional forms. The model and training settings are given in Appendix D.1.1.

4.3 Approximate Markov Chain Monte Carlo Sampler with FNLE

We obtain the approximate sampler by replacing the transition densities in Section 4.1 with FNLE densities and using SIR to sample state values at candidate times. For $j = 2, \dots, L$, define

$$\hat{e}_j(k) = \hat{p}_\phi(y_j \mid y_{j-1}, \theta_k, \Delta_j). \quad (20)$$

Algorithm 1 Continuous-time SSDE sampler with time-conditioned FNLE

Require: Data $\{(x_i, t_i^{\text{obs}})\}_{i=1}^N$

Initial $\mathcal{T}_{\text{true}}, Q, \Theta, \tau^2$; trained $\hat{p}_\phi$

- 1: Set $\mathcal{T}_{\text{all}} = \mathcal{T}_{\text{obs}} \cup \mathcal{T}_{\text{true}}$
 - 2: Initialize $y_{1:L}$ and $z_{1:L}$ on $\mathcal{T}_{\text{all}}$, consistent with $\mathcal{T}_{\text{true}}$
 - 3: Compute Ω, B (Eq. (28))
 - 4: **for** iteration $n = 1, \dots, N_{\text{iter}}$ **do**
 - 5: Sample $\mathcal{T}_{\text{cand}}$ (Section 2.2)
 - 6: Set $\mathcal{T}_{\text{event}} = \mathcal{T}_{\text{true}} \cup \mathcal{T}_{\text{cand}}$
 - 7: Set $\mathcal{T}_{\text{all}}$ (Eq. (4))
 - 8: Re-evaluate $z_{1:L}$ on the new $\mathcal{T}_{\text{all}}$
 - 9: SIR: sample $\{y_t\}_{t \in \mathcal{T}_{\text{cand}}}$
 - 10: NUTS: sample $y_{1:L}$ (Eq. (15))
 - 11: FFBS: sample $z_{1:L}$ (Eq. (16))
 - 12: Set $\mathcal{T}_{\text{true}}$ (Eq. (6))
 - 13: NUTS: sample Θ (Eq. (17))
 - 14: Sample τ^2 (Eq. (47))
 - 15: Sample $\{q_{kh}\}_{k \neq h}$ (Eq. (49))
 - 16: Set q_{kk} (Eq. (50)); recompute Ω, B
-

Replacing e_j with $\hat{e}_j$ in Eqs. (15), (16), and (17) defines the approximate conditionals on a fixed grid. The chain factorization still permits FFBS for $z_{1:L}$. For $y_{1:L}$ and Θ , we use the No-U-Turn Sampler (NUTS) [14], using gradients of the corresponding FNLE-based conditional log densities. The candidate-time, observation-noise, and transition-rate updates are unchanged.

For the state values at candidate times, SIR [31] approximates the bridge distribution in Eq. (13) with FNLE transition densities. Particles are sampled forward from the fixed left endpoint and weighted by the FNLE density at the fixed right endpoint (Appendix C.1). Thus, FNLE must support conditional sampling and differentiable density evaluation.

Algorithm Summary Algorithm 1 summarizes the proposed sampler with FNLE. Here, N_{iter} denotes the total number of MCMC sweeps. Appendix C.4 describes its per-sweep computational costs.

4.4 Approximation Properties

In our approximate MCMC sampler, FNLE approximates the transition densities, and the finite number of SIR particles introduces a separate error in sampling from the bridge distribution. Because FNLE densities do not necessarily satisfy the Chapman–Kolmogorov equation, inserting intermediate grid points can change the implied transition density. Consequently, successive sweeps apply the FNLE-based state, regime, and parameter updates on different grids and are not guaranteed to preserve a common approximate target. Exact or Chapman–Kolmogorov-consistent transition densities remove this inconsistency, but not the error due to the finite number of

SIR particles. On a fixed grid, an MH transition, such as NUTS with fixed tuning parameters, preserves its conditional target and introduces no additional target approximation [25].

We assess FNLE error through the displacement of conditional parameter modes. Consider one series with a realized combined time grid $\mathcal{G} = \mathcal{T}_{\text{all}} = \{0 = t_1 < \dots < t_L = T\}$, and condition on fixed $y_{1:L}$ and $z_{1:L}$. Let $q_{\mathcal{G}}$ and $\hat{q}_{\mathcal{G}}$ be the exact and FNLE-based versions of the unnormalized conditional density in Eq. (17). Let $\mathcal{K}$ contain the modes of both $q_{\mathcal{G}}$ and $\hat{q}_{\mathcal{G}}$. For $j = 2, \dots, L$, define the local log-density error as

$$\eta_j(\Theta) = \log \hat{p}_{\phi}(y_j | y_{j-1}, h_{z_{j-1}}(\Theta), \Delta_j) - \log p(y_j | y_{j-1}, h_{z_{j-1}}(\Theta), \Delta_j). \quad (21)$$

Write $\ell_{\mathcal{G}} = T^{-1} \log q_{\mathcal{G}}$ and $\hat{\ell}_{\mathcal{G}} = T^{-1} \log \hat{q}_{\mathcal{G}}$, and let $\mathcal{M}_{\mathcal{G}} = \arg \max_{\Theta \in \mathcal{K}} \ell_{\mathcal{G}}(\Theta)$. For a function e , let $\text{osc}_{\mathcal{K}}(e) = \sup_{\Theta \in \mathcal{K}} e - \inf_{\Theta \in \mathcal{K}} e$, and let $\text{dist}(\Theta, \mathcal{M}_{\mathcal{G}}) = \inf_{\vartheta \in \mathcal{M}_{\mathcal{G}}} \|\Theta - \vartheta\|_2$. We use oscillation because additive log-density errors independent of Θ do not affect the modes. The following proposition bounds the displacement of conditional parameter modes caused by FNLE approximation.

Proposition 1 (Stability of FNLE-based conditional parameter modes). *Assume that both $q_{\mathcal{G}}$ and $\hat{q}_{\mathcal{G}}$ attain their maxima on $\mathcal{K}$ for every $\mathcal{G}$ and its associated fixed $y_{1:L}$ and $z_{1:L}$. Choose any $\Theta_{\mathcal{G}}^* \in \mathcal{M}_{\mathcal{G}}$. Suppose that there exist $\delta \geq 0$ and $c > 0$ such that, uniformly over $\mathcal{G}$ and its associated fixed $y_{1:L}$ and $z_{1:L}$,*

$$\begin{aligned} \text{osc}_{\mathcal{K}}(\eta_j) &\leq \delta, \quad j = 2, \dots, L, \\ \ell_{\mathcal{G}}(\Theta_{\mathcal{G}}^*) - \ell_{\mathcal{G}}(\Theta) &\geq \frac{c}{2} \text{dist}(\Theta, \mathcal{M}_{\mathcal{G}})^2, \quad \Theta \in \mathcal{K}. \end{aligned} \quad (22)$$

For any $\hat{\Theta}_{\mathcal{G}} \in \arg \max_{\Theta \in \mathcal{K}} \hat{\ell}_{\mathcal{G}}(\Theta)$, set $D_{\mathcal{G}} = \text{dist}(\hat{\Theta}_{\mathcal{G}}, \mathcal{M}_{\mathcal{G}})$. Let $\mathbb{E}_{\text{unif}}[\cdot | Q]$ denote expectation over the uniformization event grid generated by a Poisson process with constant rate $\Omega(Q)$ on $[0, T]$, conditional on Q , with the observation grid fixed. Then

$$\mathbb{E}_{\text{unif}}[D_{\mathcal{G}} | Q] \leq \left[\frac{2\delta}{c} \left\{ \frac{N-1}{T} + \Omega(Q) \right\} \right]^{1/2}. \quad (23)$$

The first condition bounds parameter-dependent variation in the FNLE log-density error; the second is a quadratic separation condition requiring the normalized log posterior to decrease at least quadratically with distance from its mode set. Both conditions must hold uniformly over grids and fixed latent variables and are not guaranteed by FNLE training. This local sensitivity result does not guarantee the marginal posterior or the full sampler. At a fixed observation frequency $(N-1)/T$, the bound remains uniformly bounded as N increases, provided that $\Omega(Q)$, δ , and c^{-1} remain bounded. A proof is provided in Appendix B.

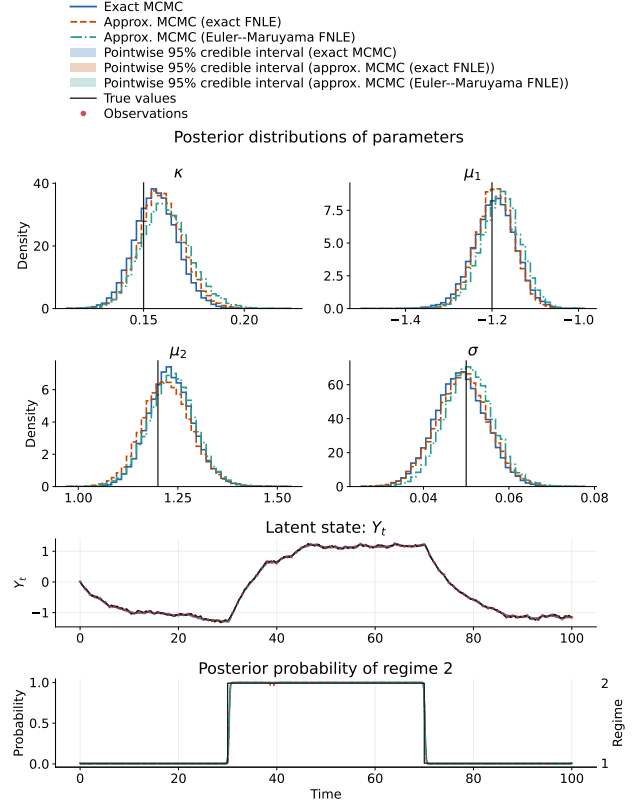


Figure 2: Comparison of inference results for the OU experiment using an exact MCMC sampler and two approximate MCMC samplers with FNLE trained on exact transitions and Euler–Maruyama simulations, respectively.

5 NUMERICAL EXPERIMENTS

MCMC settings and initialization are detailed in Appendix D.1.3. Computing infrastructure and software are described in Appendices D.5.1 and D.5.2, respectively.

5.1 Posterior Comparison on an Ornstein–Uhlenbeck Model

To assess the effect of FNLE approximation error, we compare posterior distributions obtained with two approximate MCMC samplers and an exact MCMC sampler (Section 4.1) for a tractable SSDE. We use an SSDE with $K = 2$ and an Ornstein–Uhlenbeck (OU) model [33] in each regime. In regime k , the model is

$$dY_t = \kappa(\mu_k - Y_t)dt + \sigma dW_t. \quad (24)$$

Here, W_t is a one-dimensional standard Brownian motion. Each regime-specific OU SDE has an exact transition density and bridge distribution, making this SSDE tractable under the definition in Sec

Table 2: Characteristics of the SSDE Models Used to Generate the Synthetic Datasets

Model	Observation error	Nonlinear drift	State-dependent diffusion	Intractable transition density	Multivariate states
Ornstein–Uhlenbeck	✓	✗	✗	✗	✗
Lotka–Volterra	✓	✓	✓	✓	✓
Gene-expression CLE	✓	✗	✓	✓	✓
Susceptible–infected–recovered epidemic model	✓	✓	✓	✓	✓

✓: Present; ✗: Absent.

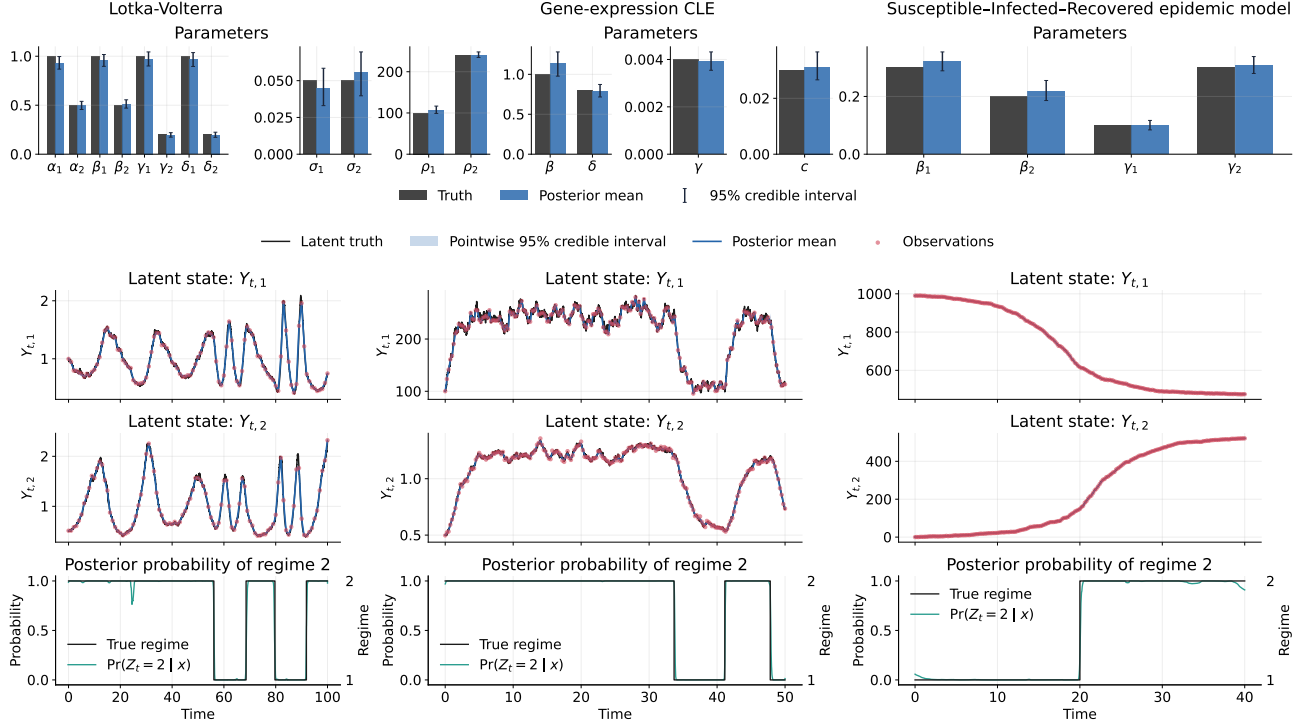


Figure 3: Inference results for synthetic data from three SSDEs with regime-specific Lotka–Volterra, gene-expression chemical Langevin, and susceptible–infected–recovered epidemic models (left to right).

tion 4. Figure 2 compares posterior parameter distributions from the three samplers. The sampler using FNLE trained on exact transitions produces posterior marginals close to those from the exact sampler. The marginals remain similar when FNLE is trained on Euler–Maruyama simulations instead, as in the subsequent experiments on intractable SSDEs. All three samplers recovered the SDE parameters and regime path, with $\hat{R}$ values close to one (Appendix Table 10). The experimental settings are given in Appendix D.2.1.

5.2 Synthetic-Data Experiments

We evaluate regime-path recovery and estimation of Θ using synthetic data from three SSDE models with

$K = 2$ and known regimes and parameters. The remaining three rows of Table 2 summarize these intractable SSDEs; the first row describes the tractable OU benchmark in Section 5.1.

All three models include observation error, state-dependent diffusion, and multivariate latent states. The previous methods in Table 1 have not demonstrated applications combining these features.

$W_{t,1}$ and $W_{t,2}$ denote independent Brownian motions. Complete settings of these three experiments are given in Appendices D.2.2, D.2.3, and D.2.4, respectively.

Lotka–Volterra Equation We write $Y_{t,1}$ and $Y_{t,2}$ for predator and prey abundance, respectively. The drift follows the classical Lotka–Volterra (LV)

predator–prey model [26, 35], and independent multiplicative noise is introduced as in Li and Mao [23]. In regime k , the model is

$$\begin{aligned} dY_{t,1} &= (\gamma_k Y_{t,2} Y_{t,1} - \delta_k Y_{t,1}) dt \\ &\quad + \sigma_1 Y_{t,1} dW_{t,1}, \\ dY_{t,2} &= (\alpha_k Y_{t,2} - \beta_k Y_{t,2} Y_{t,1}) dt \\ &\quad + \sigma_2 Y_{t,2} dW_{t,2}. \end{aligned} \quad (25)$$

Chemical Langevin Equation We write $Y_{t,1}$ for the mRNA copy number and $Y_{t,2}$ for the fluorescence intensity proportional to the protein copy number, using continuous approximations to the copy numbers. We use a chemical Langevin equation (CLE) [10], whose dynamics in regime k are

$$\begin{aligned} dY_{t,1} &= (\alpha_k - \beta Y_{t,1}) dt \\ &\quad + \sqrt{\alpha_k + \beta Y_{t,1}} dW_{t,1}, \\ dY_{t,2} &= (\gamma Y_{t,1} - \delta Y_{t,2}) dt \\ &\quad + c \sqrt{\gamma Y_{t,1} + \delta Y_{t,2}} dW_{t,2}. \end{aligned} \quad (26)$$

For inference, we reparameterize the CLE using $\rho_k = \alpha_k/\beta$ and β instead of α_k and β to improve MCMC mixing. The coefficient c^2 converts protein copy number to fluorescence intensity.

Susceptible–Infected–Recovered Epidemic Model We write $Y_{t,1}$ and $Y_{t,2}$ for the susceptible and recovered populations, respectively. The drift follows the classical susceptible–infected–recovered epidemic model [16], and the diffusion terms follow the chemical Langevin approximation [10]. The infected population is determined by $I_t = N_{\text{pop}} - Y_{t,1} - Y_{t,2}$, where N_{pop} is the fixed total population. In regime k , the model is

$$\begin{aligned} dY_{t,1} &= -\frac{\beta_k Y_{t,1} I_t}{N_{\text{pop}}} dt - \sqrt{\frac{\beta_k Y_{t,1} I_t}{N_{\text{pop}}}} dW_{t,1}, \\ dY_{t,2} &= \gamma_k I_t dt + \sqrt{\gamma_k I_t} dW_{t,2}. \end{aligned} \quad (27)$$

Posterior means were close to the true parameters with narrow 95% credible intervals in all three experiments (Figure 3). The posterior regime probabilities also recovered the true switching pattern. The τ_1 parameter in the CLE experiment and τ_1, τ_2 in the susceptible–infected–recovered experiment showed poor mixing (Appendix Tables 12 and 13); diagnostics for the LV experiment are in Appendix Table 11.

5.3 Real-Data Application

We applied the proposed method to 71 paired abundance observations of *Didinium* and *Paramecium* from Villeux [34], using the LV SSDE in Eq. (25) with

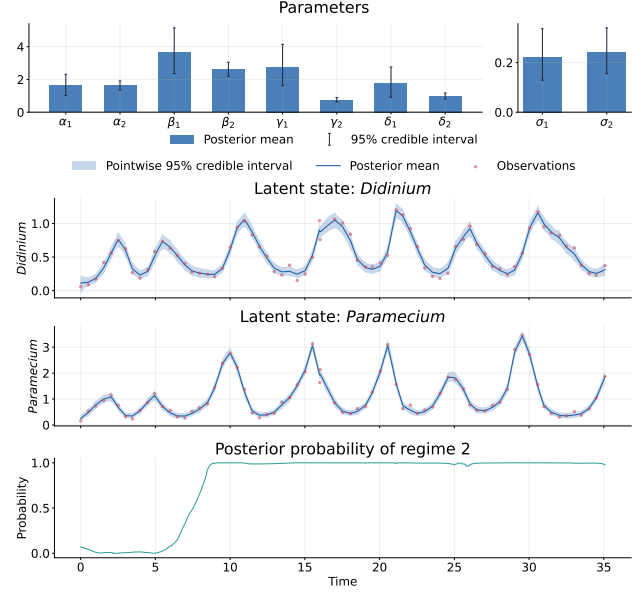


Figure 4: Real-data inference results for the *Didinium*–*Paramecium* time series using a Lotka–Volterra SSDE.

$K = 2$. Data preprocessing and inference settings are given in Appendix D.3.

As shown in Figure 4, the posterior probability of regime 2 increased around $t = 7$ and remained close to one thereafter. Notably, the initial regime identified in an unsupervised manner closely corresponds to the transient period that was manually excluded prior to fitting a LV model in a previous analysis of these data [24]. Appendix Figure 5 shows similar chain-specific marginal posterior distributions and overlapping traces across the four chains. The diagnostics in Appendix Table 14 also suggest convergence.

6 CONCLUSION

We proposed a broadly applicable approximate MCMC sampler combining time-conditioned FNLE and uniformization for joint inference of SSDE latent states, regimes, and parameters. Starting from an exact uniformization-based MCMC sampler for tractable SSDEs, we used FNLE to approximate transition densities and SIR to sample approximately from bridge distributions, extending the approach to intractable SSDEs. In the OU benchmark, approximate samplers using FNLE trained on exact or Euler–Maruyama transitions produced posterior parameter marginals similar to those from exact MCMC. The approximate sampler also recovered parameters and regime paths for the three SSDE models, to which existing methods have limited applicability. In the real-data analysis

of the *Didinium-Paramecium* time series, the method identified a persistent regime transition from the initial transient dynamics to the subsequent predator-prey dynamics.

AI use statement

In this work, we used generative AI tools to help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, design or provide feedback on research methodology or experiments, implement methods, and assist with translation. We have not used generative AI tools to generate synthetic datasets, propose or refine hypotheses, clean and reformat datasets, support qualitative and thematic data analysis, or interpret results. Additionally, we used generative AI tools to create or modify scientific figures or images, suggest experimental parameters, create or edit software code, draft parts of a research paper, summarize or analyze existing literature, brainstorm, source or search for information, edit a research paper to improve readability, identify relevant literature, and propose a title or keywords. We have reviewed all AI-assisted work. We manually reviewed all LLM-generated code, draft text, and proofs. All changes, whether AI-assisted or manual, were tracked using `git`, and we inspected the corresponding diffs to carefully review modifications made with the assistance of LLMs. LLM-assisted proofs were checked by the authors for correctness. All references were added to the BibTeX file manually after verifying their bibliographic information against the original sources and reading the cited works to confirm their content and relevance. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

Code Availability

The code for the numerical experiments is available at <https://github.com/shion-h/fnle-ssde>.

References

- [1] Eli Bingham, Jonathan P. Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul A. Szerlip, Paul Horsfall, and Noah D. Goodman. Pyro: Deep universal probabilistic programming. *J. Mach. Learn. Res.*, 20:28:1–28:6, 2019. URL <http://jmlr.org/papers/v20/18-403.html>.
- [2] P. G. Blackwell. Bayesian inference for Markov processes with diffusion and discrete components. *Biometrika*, 90(3):613–627, 09 2003. ISSN 0006-3444. doi: 10.1093/biomet/90.3.613. URL <https://doi.org/10.1093/biomet/90.3.613>.
- [3] Paul G. Blackwell, Mu Niu, Mark S. Lambert, and Scott D. LaPoint. Exact Bayesian inference for animal movement in continuous time. *Methods in Ecology and Evolution*, 7(2):184–195, 2016. doi: <https://doi.org/10.1111/2041-210X.12460>. URL <https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12460>.
- [4] Jan Boelts, Michael Deistler, Manuel Gloeckler, Álvaro Tejero-Cantero, Jan-Matthis Lueckmann, Guy Moss, Peter Steinbach, Thomas Moreau, Fabio Muratore, Julia Linhart, Conor Durkan, Julius Vetter, Benjamin Kurt Miller, Maternus Herold, Abolfazl Ziaemehr, Matthijs Pals, Theo Gruner, Sebastian Bischoff, Nastya Krouglova, Richard Gao, Janne K. Lappalainen, Bálint Mucsányi, Felix Pei, Auguste Schulz, Zinovia Stefanidi, Pedro Rodrigues, Cornelius Schröder, Faried Abu Zaid, Jonas Beck, Jaivardhan Kapoor, David S. Greenberg, Pedro J. Gonçalves, and Jakob H. Macke. sbi reloaded: a toolkit for simulation-based inference workflows. *Journal of Open Source Software*, 10(108):7754, 2025. doi: 10.21105/joss.07754. URL <https://doi.org/10.21105/joss.07754>.
- [5] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020. doi: 10.1073/pnas.1912789117. URL <https://www.pnas.org/doi/abs/10.1073/pnas.1912789117>.
- [6] Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2019. Curran Associates Inc.
- [7] David A. Van Dyk and Taeyoung Park. Partially collapsed Gibbs samplers: Theory and methods. *Journal of the American Statistical Association*, 103(482):790–796, 2008. ISSN 01621459. URL <http://www.jstor.org/stable/27640101>.
- [8] S. Frühwirth-Schnatter. *Finite Mixture and Markov Switching Models*. Springer Series in Statistics. Springer New York, 2006. ISBN 9780387357683. URL <https://books.google.ne/books?id=f8KiI7eRjYoC>.
- [9] Anastasis Georgoulas, Jane Hillston, and Guid Sanguinetti. ProPPA: Probabilistic programming for stochastic dynamical systems. *ACM Trans. Model. Comput. Simul.*, 28(1), January 2018. ISSN 1049-3301. doi: 10.1145/3154392. URL <https://doi.org/10.1145/3154392>.
- [10] Daniel T. Gillespie. The chemical Langevin equation. *The Journal of Chemical Physics*, 113(1): 297–306, 07 2000. ISSN 0021-9606. doi: 10.1063/1.481811. URL <https://doi.org/10.1063/1.481811>.
- [11] Manuel Gloeckler, Shoji Toyota, Kenji Fukumizu,

- and Jakob H. Macke. Compositional simulation-based inference for time series. In *The Thirteenth International Conference on Learning Representations*, 2025. URL <https://openreview.net/forum?id=uC1UJk05H>.
- [12] Flávio B Gonçalves, Lívia M Dutra, and Roger WC Silva. Exact and computationally efficient Bayesian inference for generalized Markov modulated Poisson processes. *Statistics and Computing*, 32(1):14, 2022.
- [13] El Houcine Hibbah, Hamid El Maroufy, Christiane Fuchs, and Taib Ziad. An MCMC computational approach for a continuous time state-dependent regime switching diffusion process. *Journal of Applied Statistics*, 47(8):1354–1374, 2020. doi: 10.1080/02664763.2019.1677573. URL <https://doi.org/10.1080/02664763.2019.1677573>. PMID: 35706700.
- [14] Matthew D. Hoffman and Andrew Gelman. The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15(47):1593–1623, 2014. ISSN 1533-7928.
- [15] Shion Hosoda, Tsukasa Fukunaga, and Michiaki Hamada. Umibato: Estimation of time-varying microbial interaction using continuous-time regression hidden Markov model. *Bioinformatics*, 37(Supplement.1):i16–i24, July 2021. ISSN 1367-4803. doi: 10.1093/bioinformatics/btab287.
- [16] William Ogilvy Kermack and A. G. McKendrick. A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 115(772):700–721, 08 1927. ISSN 0950-1207. doi: 10.1098/rspa.1927.0118. URL <https://doi.org/10.1098/rspa.1927.0118>.
- [17] Naoki Kiyohara, Edward Johns, and Yingzhen Li. Neural stochastic flows: Solver-free modelling and inference for SDE solutions. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen, editors, *Advances in Neural Information Processing Systems*, volume 38, pages 16917–16962. Curran Associates, Inc., 2025. doi: 10.52202/085713-0571. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/18abbeef8cfe9203fdf9053c9c4fe191-Paper-Conference.pdf.
- [18] P.E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Stochastic Modelling and Applied Probability. Springer Berlin Heidelberg, 2013. ISBN 9783662126165. URL <https://books.google.co.jp/books?id=r9r6CAAQBAJ>.
- [19] Lukas Köhs, Bastian Alt, and Heinz Koepl. Variational Inference for Continuous-Time Switching Dynamical Systems. In *Advances in Neural Information Processing Systems*, volume 34, pages 20545–20557. Curran Associates, Inc., 2021.
- [20] Lukas Köhs, Bastian Alt, and Heinz Koepl. Markov Chain Monte Carlo for Continuous-Time Switching Dynamical Systems. In *Proceedings of the 39th International Conference on Machine Learning*, pages 11430–11454. PMLR, June 2022.
- [21] Ravin Kumar, Colin Carroll, Ari Hartikainen, and Osvaldo Martin. Arviz a unified library for exploratory analysis of bayesian models in python. *Journal of Open Source Software*, 4(33):1143, 2019. doi: 10.21105/joss.01143. URL <https://doi.org/10.21105/joss.01143>.
- [22] Haitao Li, Tao Li, and Cindy Yu. No-arbitrage Taylor rules with switching regimes. *Management Science*, 59(10):2278–2294, 2013. doi: 10.1287/mnsc.1120.1702. URL <https://doi.org/10.1287/mnsc.1120.1702>.
- [23] Xiaoyue Li and Xuerong Mao. Population dynamical behavior of non-autonomous Lotka-Volterra competitive system with random perturbation. *Discrete and Continuous Dynamical Systems*, 24(2):523–545, 2009.
- [24] Chen Liao, Joao B Xavier, and Zhenduo Zhu. Enhanced inference of ecological networks by parameterizing ensembles of population dynamics models constrained with prior knowledge. *BMC ecology*, 20(1):3, 2020.
- [25] Jun S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer, 2001.
- [26] Alfred J. Lotka. Analytical note on certain rhythmic relations in organic systems. *Proceedings of the National Academy of Sciences*, 6(7):410–415, 1920. doi: 10.1073/pnas.6.7.410. URL <https://www.pnas.org/doi/abs/10.1073/pnas.6.7.410>.
- [27] X. Mao and C. Yuan. *Stochastic Differential Equations with Markovian Switching*. G - Reference, Information and Interdisciplinary Subjects Series. Imperial College Press, 2006. ISBN 9781860947018. URL <https://books.google.co.jp/books?id=mEMMAFGfr1sC>.
- [28] Jiangwei Pan, Vinayak Rao, Pankaj Agarwal, and Alan Gelfand. Markov-modulated marked Poisson processes for check-in data. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 2244–2253, New York, New York, USA, 20–22 Jun 2016.

- PMLR. URL <https://proceedings.mlr.press/v48/pana16.html>.
- [29] George Papamakarios, David Sterratt, and Iain Murray. Sequential Neural Likelihood: Fast Likelihood-free Inference with Autoregressive Flows. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, pages 837–848. PMLR, April 2019.
 - [30] Vinayak Rao and Yee Whye Teh. Fast MCMC Sampling for Markov Jump Processes and Extensions. *Journal of Machine Learning Research*, 14 (103):3295–3320, 2013. ISSN 1533-7928.
 - [31] Donald B. Rubin. The calculation of posterior distributions by data augmentation: Comment: A noniterative sampling/importance resampling alternative to the data augmentation algorithm for creating a few imputations when fractions of missing information are modest: The SIR algorithm. *Journal of the American Statistical Association*, 82(398):543–546, 1987. ISSN 01621459, 1537274X. URL <http://www.jstor.org/stable/2289460>.
 - [32] Timothée Stumpf-Fétizon, Krzysztof Łatuszyński, Jan Palczewski, and Gareth Roberts. Exact Bayesian inference for Markov switching diffusions. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, page qkag115, July 2026. ISSN 1369-7412. doi: 10.1093/jrsssb/qkag115.
 - [33] G. E. Uhlenbeck and L. S. Ornstein. On the theory of the brownian motion. *Phys. Rev.*, 36:823–841, Sep 1930. doi: 10.1103/PhysRev.36.823. URL <https://link.aps.org/doi/10.1103/PhysRev.36.823>.
 - [34] B. G. Veilleux. An analysis of the predatory interaction between Paramecium and Didinium. *Journal of Animal Ecology*, 48(3):787–803, 1979. ISSN 00218790, 13652656. URL <http://www.jstor.org/stable/4195>.
 - [35] Vito Volterra. Fluctuations in the abundance of a species considered mathematically. *Nature*, 119: 12–13, 1927.
 - [36] Jun Yan, Yung-wei Chen, Kirstin Lawrence-Apfel, Isaac M. Ortega, Vladimir Pozdnyakov, Scott Williams, and Thomas Meyer. A moving–resting process with an embedded Brownian motion for animal movements. *Population Ecology*, 56(2):401–415, 2014. doi: <https://doi.org/10.1007/s10144-013-0428-8>. URL <https://esj-journals.onlinelibrary.wiley.com/doi/abs/10.1007/s10144-013-0428-8>.

Broadly Applicable Approximate MCMC for Switching Stochastic Differential Equations Using Uniformization and Time-Conditioned Factorized Neural Likelihood Estimation:

Supplementary Materials

A EXACT SAMPLING CONSTRUCTION

We derive an exact version of Algorithm 1 from the proper full-path posterior in Eq. (12), assuming exact SDE transition densities and sampling from bridge distributions are available. For any time set $\mathcal{G}$, define $Y_{\mathcal{G}} = \{Y_t\}_{t \in \mathcal{G}}$ and $Y_{\setminus \mathcal{G}} = \{Y_t\}_{t \in [0, T] \setminus \mathcal{G}}$, with realizations denoted by lowercase symbols, and let $\mathcal{T}_{\text{ret}} = \mathcal{T}_{\text{obs}} \cup \mathcal{T}_{\text{true}}$. Each sweep first samples $(\mathcal{T}_{\text{cand}}, Y_{\setminus \mathcal{T}_{\text{ret}}})$ conditional on the current regime, retained state values, and parameters. With the resulting $\mathcal{T}_{\text{event}}$ and $\mathcal{T}_{\text{all}}$ fixed, it updates the blocks Y , $(Z, Y_{\setminus \mathcal{T}_{\text{all}}})$, and $(\Theta, Y_{\setminus \mathcal{T}_{\text{all}}})$, followed by τ^2 . The candidate-time augmentation is then marginalized before updating Q . The factorizations below permit trimming unused path draws by partially collapsed Gibbs sampling [7].

A.1 Uniformization Matrices and Candidate Times

For a transition rate matrix Q , choose a deterministic rate $\Omega = \Omega(Q) > \max_k(-q_{kk})$. The uniformized chain has transition matrix [30]

$$B = I_K + \frac{Q}{\Omega}, \quad (28)$$

where I_K is the $K \times K$ identity matrix. On $\mathcal{T}_{\text{all}}$, event times permit regime transitions, whereas observation-only times preserve the regime:

$$A_j = \begin{cases} B, & t_j \in \mathcal{T}_{\text{event}}, \\ I_K, & t_j \in \mathcal{T}_{\text{obs}}, \end{cases} \quad j = 2, \dots, L. \quad (29)$$

Event times almost surely do not coincide with the fixed observation times. See Rao and Teh [30] for the uniformization construction and its MCMC application.

For candidate generation, let $[a_r, b_r]$, $r = 1, \dots, R_Z$, be the intervals between consecutive times in $\{0, T\} \cup \mathcal{T}_{\text{true}}$, with regime k_r in each interval's interior. Conditional on the current path and Q , sample independently across intervals

$$M_r \sim \text{Poisson}\{(\Omega + q_{k_r, k_r})(b_r - a_r)\}, \quad (30)$$

followed by

$$u_{r,m} \mid M_r \stackrel{\text{i.i.d.}}{\sim} \text{Uniform}(a_r, b_r), \quad m = 1, \dots, M_r, \quad (31)$$

where $\text{Poisson}(\lambda)$ has mean λ , and $\text{Uniform}(a, b)$ is uniform on (a, b) . The sampled times $u_{r,m}$ form $\mathcal{T}_{\text{cand}}$.

A.2 Block Gibbs Construction

A.2.1 Candidate Grids and Path Augmentation

Conditional on Z , $\mathcal{T}_{\text{ret}}$ is fixed, and the $(\mathcal{T}_{\text{cand}}, Y_{\setminus \mathcal{T}_{\text{ret}}})$ update factors by conditional independence:

$$\begin{aligned} p(\mathcal{T}_{\text{cand}}, y_{\setminus \mathcal{T}_{\text{ret}}} \mid y_{\mathcal{T}_{\text{ret}}}, z, \Theta, Q, \tau^2, x) \\ = p(\mathcal{T}_{\text{cand}} \mid z, Q) \\ \times p(y_{\setminus \mathcal{T}_{\text{ret}}} \mid y_{\mathcal{T}_{\text{ret}}}, z, \Theta). \end{aligned} \quad (32)$$

The first factor is the conditional Poisson process in Eqs. (30) and (31) [30].

Use the local indices $a : b$ from Eq. (13), with endpoints t_{a-1} and t_{b+1} consecutive in $\mathcal{T}_{\text{ret}}$ and regime k between them. By the Markov property and the Chapman–Kolmogorov equation, its candidate-time marginal is

$$\begin{aligned} p(y_{a:b} \mid y_{a-1}, y_{b+1}, z, \mathcal{T}_{\text{all}}, \Theta) \\ = \frac{\prod_{j=a}^{b+1} p(y_j \mid y_{j-1}, \theta_k, \Delta_j)}{p(y_{b+1} \mid y_{a-1}, \theta_k, t_{b+1} - t_{a-1})}. \end{aligned} \quad (33)$$

Sampling the intervening paths from their bridge distributions conditional on $y_{\mathcal{T}_{\text{cand}}}$ yields $Y_{\setminus \mathcal{T}_{\text{ret}}}$.

A.2.2 Latent-State Update

The Y block factors as

$$\begin{aligned} p(y_{\mathcal{T}_{\text{all}}}, y_{\setminus \mathcal{T}_{\text{all}}} \mid z, \mathcal{T}_{\text{event}}, \Theta, \tau^2, x) \\ = p(y_{\mathcal{T}_{\text{all}}} \mid z, \mathcal{T}_{\text{event}}, \Theta, \tau^2, x) \\ \times p(y_{\setminus \mathcal{T}_{\text{all}}} \mid y_{\mathcal{T}_{\text{all}}}, z, \mathcal{T}_{\text{event}}, \Theta). \end{aligned} \quad (34)$$

Since z_{j-1} is the regime on $[t_{j-1}, t_j)$, the Markov property gives

$$\begin{aligned} p(y_{1:L} \mid z_{1:L}, \mathcal{T}_{\text{event}}, \Theta) \\ = p(y_1) \prod_{j=2}^L p(y_j \mid y_{j-1}, \theta_{z_{j-1}}, \Delta_j). \end{aligned} \quad (35)$$

Multiplying by $p(x \mid y_{1:L}, \tau^2)$ gives Eq. (15).

A.2.3 Regime Conditional Distribution

The $(Z, Y_{\setminus \mathcal{T}_{\text{all}}})$ block factors as

$$\begin{aligned} p(z, y_{\setminus \mathcal{T}_{\text{all}}} \mid y_{\mathcal{T}_{\text{all}}}, \mathcal{T}_{\text{event}}, \Theta, Q, \tau^2, x) \\ = p(z \mid y_{\mathcal{T}_{\text{all}}}, \mathcal{T}_{\text{event}}, \Theta, Q, \tau^2, x) \\ \times p(y_{\setminus \mathcal{T}_{\text{all}}} \mid y_{\mathcal{T}_{\text{all}}}, z, \mathcal{T}_{\text{event}}, \Theta). \end{aligned} \quad (36)$$

Uniformization [30] and identity transitions at observation-only times give

$$\begin{aligned} p(\mathcal{T}_{\text{event}}, z_{1:L} \mid Q) \\ = p(\mathcal{T}_{\text{event}} \mid \Omega(Q)) p(z_1) \prod_{j=2}^L A_j(z_{j-1}, z_j). \end{aligned} \quad (37)$$

Here, $p(\mathcal{T}_{\text{event}} \mid \Omega(Q))$ is the Poisson-process density. Bayes' rule with Eqs. (37) and (35) expresses the first factor in Eq. (36) as Eq. (16).

A.2.4 Parameter Updates

The $(\Theta, Y_{\setminus \mathcal{T}_{\text{all}}})$ block factors as

$$\begin{aligned} p(\Theta, y_{\setminus \mathcal{T}_{\text{all}}} \mid y_{\mathcal{T}_{\text{all}}}, z, \mathcal{T}_{\text{event}}, Q, \tau^2, x) \\ = p(\Theta \mid y_{\mathcal{T}_{\text{all}}}, z, \mathcal{T}_{\text{event}}) \\ \times p(y_{\setminus \mathcal{T}_{\text{all}}} \mid y_{\mathcal{T}_{\text{all}}}, z, \mathcal{T}_{\text{event}}, \Theta). \end{aligned} \quad (38)$$

Using Eq. (35) and the independent prior for Θ gives Eq. (17) for the first factor. The τ^2 update in Eq. (47) depends on Y only through $Y_{\mathcal{T}_{\text{obs}}}$.

Marginalizing $\mathcal{T}_{\text{cand}}$ recovers Eq. (12), under which $Q \mid z$ follows Eq. (49). After this collapsed update, new candidate times must be generated using the updated Q before any further grid-dependent update [7].

A.2.5 Trimming and Invariance Across Iterations

Within a sweep, each draw of $Y_{\mathcal{T}_{\text{all}}}$ is unused by the subsequent marginal update; the τ^2 and Q updates also require no such values.

Every updated true-jump time lies in $\mathcal{T}_{\text{event}}$, implying that the updated $\mathcal{T}_{\text{ret}}$ is a subset of the current $\mathcal{T}_{\text{all}}$. Equation (32) resamples the path outside this retained set before any values at new candidate times are used. Thus, the final intervening-path draw of the previous sweep is also unused.

Trimming these draws preserves the retained-variable transition kernel [7]. The block updates and augmentation preserve the full-path posterior marginal; hence, at sweep boundaries, the invariant distribution is the marginal of Eq. (12) on

$$(Z, Y_{\mathcal{T}_{\text{ret}}}, \Theta, \tau^2, Q), \quad \mathcal{T}_{\text{ret}} = \mathcal{T}_{\text{obs}} \cup \mathcal{T}_{\text{true}}. \quad (39)$$

In particular, its (Z, Θ, τ^2, Q) marginal is the desired posterior marginal.

For fixed $\mathcal{T}_{\text{event}}$, the retained conditionals share the same marginalized posterior. The $Y_{\mathcal{T}_{\text{all}}}$ and Θ draws may therefore be replaced by Markov transitions preserving their respective conditionals, including NUTS with fixed tuning [14, 25].

B PROOF OF PROPOSITION 1

Proof. Fix a realized grid $\mathcal{G}$ and its associated $y_{1:L}$ and $z_{1:L}$. Because the exact and FNLE-based conditional densities differ only in their transition-density factors, they can be written with a common multiplicative convention as

$$\begin{aligned} q_{\mathcal{G}}(\Theta) &= p(\Theta) \prod_{j=2}^L p(y_j \mid y_{j-1}, h_{z_{j-1}}(\Theta), \Delta_j), \\ \hat{q}_{\mathcal{G}}(\Theta) &= p(\Theta) \prod_{j=2}^L \hat{p}_{\phi}(y_j \mid y_{j-1}, h_{z_{j-1}}(\Theta), \Delta_j). \end{aligned}$$

Any omitted factors are independent of Θ and therefore do not affect either conditional mode. The common prior factor cancels from the difference between the normalized log densities, and we have

$$\begin{aligned} e_{\mathcal{G}}(\Theta) &:= \hat{\ell}_{\mathcal{G}}(\Theta) - \ell_{\mathcal{G}}(\Theta) \\ &= \frac{1}{T} \sum_{j=2}^L \eta_j(\Theta). \end{aligned}$$

For arbitrary functions $f_2, \dots, f_L$ on $\mathcal{K}$,

$$\text{osc}_{\mathcal{K}} \left(\sum_{j=2}^L f_j \right) \leq \sum_{j=2}^L \text{osc}_{\mathcal{K}}(f_j).$$

Applying this inequality to the local errors and using the first condition in Eq. (22) yields

$$\begin{aligned} \text{osc}_{\mathcal{K}}(e_{\mathcal{G}}) &\leq \frac{1}{T} \sum_{j=2}^L \text{osc}_{\mathcal{K}}(\eta_j) \\ &\leq \frac{(L-1)\delta}{T}. \end{aligned} \quad (40)$$

The maximizing property of $\hat{\Theta}_{\mathcal{G}}$ gives

$$\hat{\ell}_{\mathcal{G}}(\hat{\Theta}_{\mathcal{G}}) \geq \hat{\ell}_{\mathcal{G}}(\Theta_{\mathcal{G}}^*).$$

Substituting $\hat{\ell}_G = \ell_G + e_G$ and rearranging gives

$$\begin{aligned}\ell_G(\Theta_G^*) - \ell_G(\hat{\Theta}_G) &\leq e_G(\hat{\Theta}_G) - e_G(\Theta_G^*) \\ &\leq \text{osc}_{\mathcal{K}}(e_G).\end{aligned}$$

Combining the second condition in Eq. (22) with Eq. (40) therefore gives

$$\begin{aligned}\frac{c}{2}D_G^2 &\leq \ell_G(\Theta_G^*) - \ell_G(\hat{\Theta}_G) \\ &\leq \frac{(L-1)\delta}{T}.\end{aligned}$$

Consequently, every realized grid satisfies the pointwise bound

$$D_G \leq \left\{ \frac{2(L-1)\delta}{cT} \right\}^{1/2}. \quad (41)$$

It remains to average Eq. (41) over the joint uniformization construction. Let $M = |\mathcal{T}_{\text{event}}|$ be the number of uniformization events. Conditional on Q , the event process is a Poisson process with rate $\Omega(Q)$, and hence

$$M \mid Q \sim \text{Poisson}\{\Omega(Q)T\}.$$

The Poisson event times are almost surely distinct from one another and from the N fixed observation times, and $L = N + M$ almost surely and

$$\mathbb{E}_{\text{unif}}[L-1 \mid Q] = N-1 + \Omega(Q)T. \quad (42)$$

The pointwise bound in Eq. (41) is uniform in the associated fixed y and z , and only the distribution of L is needed below. Taking conditional expectations in Eq. (41) and applying Jensen's inequality to the concave square-root function gives

$$\begin{aligned}\mathbb{E}_{\text{unif}}[D_G \mid Q] &\leq \left(\frac{2\delta}{cT} \right)^{1/2} \mathbb{E}_{\text{unif}}[(L-1)^{1/2} \mid Q] \\ &\leq \left\{ \frac{2\delta}{cT} \mathbb{E}_{\text{unif}}[L-1 \mid Q] \right\}^{1/2} \\ &= \left[\frac{2\delta}{c} \left\{ \frac{N-1}{T} + \Omega(Q) \right\} \right]^{1/2},\end{aligned}$$

where the final equality follows from Eq. (42). This is Eq. (23) and completes the proof. $\square$

C ALGORITHMIC DETAILS AND EXTENSIONS

C.1 Sampling State Values at Candidate Times with SIR

Before applying the NUTS update to y , we use sampling-importance-resampling (SIR) to assign latent values $\{y_t\}_{t \in \mathcal{T}_{\text{cand}}}$. Using local indices on the combined time grid containing the newly sampled $\mathcal{T}_{\text{cand}}$, consider a consecutive block $y_{a:b}$ of current state values at candidate times with fixed neighboring values y_{a-1} and y_{b+1} at adjacent points of $\mathcal{T}_{\text{obs}} \cup \mathcal{T}_{\text{true}}$. For particle $p = 1, \dots, P$, we simulate

$$y_j^{(p)} \sim \hat{p}_\phi(\cdot \mid y_{j-1}^{(p)}, \theta_{z_{j-1}}, \Delta_j), \quad j = a, \dots, b, \quad (43)$$

with $y_{a-1}^{(p)} = y_{a-1}$. Because the proposal is the forward FNLE transition-density approximation from the left endpoint, the SIR weight is

$$\tilde{w}_p = \hat{p}_\phi(y_{b+1} \mid y_b^{(p)}, \theta_{z_b}, \Delta_{b+1}), \quad (44)$$

$$w_p = \frac{\tilde{w}_p}{\sum_{q=1}^P \tilde{w}_q}. \quad (45)$$

We then draw $A \sim \text{Categorical}(w_1, \dots, w_P)$ and set $y_{a:b} = y_{a:b}^{(A)}$. This construction is applied to every interval between consecutive points of $\mathcal{T}_{\text{obs}} \cup \mathcal{T}_{\text{true}}$ that contains at least one current candidate time.

C.2 Conjugate Parameter Updates

Let τ_d^2 denote the d -th element of τ^2 . Assume independent priors $\tau_d^2 \sim \text{InvGamma}(\alpha_d^\tau, \beta_d^\tau)$ and $q_{kh} \sim \text{Gamma}(\alpha^Q, \beta^Q)$ for $k \neq h$, with the Gamma distribution parameterized by shape and rate. Here, $\text{InvGamma}(a, b)$ denotes the inverse-Gamma distribution with shape a and scale b , whose density is proportional to $v^{-a-1} \exp(-b/v)$ for $v > 0$. Let

$$R_d = \sum_{i=1}^N \left(x_{i,d} - y_{t_i^{\text{obs}},d} \right)^2. \quad (46)$$

The conjugate observation-noise update used in Algorithm 1 is

$$\tau_d^2 \mid x, y \sim \text{InvGamma} \left(\alpha_d^\tau + \frac{N}{2}, \beta_d^\tau + \frac{R_d}{2} \right). \quad (47)$$

Let n_{kh} be the number of true jumps from k to h , and let

$$m_k = \int_0^T \mathbb{I}\{Z_t = k\} dt \quad (48)$$

be the total holding time in regime k . After marginalizing the virtual jumps, the conjugate update is

$$q_{kh} \mid z \sim \text{Gamma}(\alpha^Q + n_{kh}, \beta^Q + m_k), \quad k \neq h. \quad (49)$$

The diagonal entries are set to

$$q_{kk} = - \sum_{h \neq k} q_{kh}. \quad (50)$$

C.3 Extension to Multiple Time Series

Suppose that S time series are conditionally independent given the shared parameters Θ, τ^2, Q . Series s has its own observations, latent state process, regime path, and combined time grid, denoted by $x^{(s)}, Y^{(s)}, Z^{(s)}$, and $\mathcal{T}_{\text{all}}^{(s)}$, respectively. Let N_s be its number of observations, L_s its number of combined grid points, and $[0, T_s]$ its observation interval. We use $y_j^{(s)}, z_j^{(s)}$, and $\Delta_j^{(s)}$ for its latent states and regimes at the grid points, and its interval lengths, respectively.

The initialization and the candidate-grid, latent-state, and regime updates in Algorithm 1 are performed separately for each series. The shared parameters are then updated using the contributions from all series, with each prior included only once. The FNLE-based conditional density corresponding to Eq. (17) becomes

$$\hat{p}(\Theta \mid \{y^{(s)}, z^{(s)}, \mathcal{T}_{\text{all}}^{(s)}\}_{s=1}^S) \propto p(\Theta) \prod_{s=1}^S \prod_{j=2}^{L_s} \hat{p}_\phi(y_j^{(s)} \mid y_{j-1}^{(s)}, h_{z_{j-1}^{(s)}}(\Theta), \Delta_j^{(s)}). \quad (51)$$

For the observation-noise update in Eq. (47), replace N by $\sum_{s=1}^S N_s$ and R_d by $\sum_{s=1}^S R_d^{(s)}$, where

$$R_d^{(s)} = \sum_{i=1}^{N_s} \left(x_{i,d}^{(s)} - y_{t_i^{\text{obs},(s)},d}^{(s)} \right)^2 \quad (52)$$

is the residual sum of squares for component d in series s . For Eq. (49), replace n_{kh} and m_k by $\sum_{s=1}^S n_{kh}^{(s)}$ and $\sum_{s=1}^S m_k^{(s)}$, where $n_{kh}^{(s)}$ counts the true jumps from k to h in series s , and

$$m_k^{(s)} = \int_0^{T_s} \mathbb{I}\{Z_t^{(s)} = k\} dt \quad (53)$$

is its holding time in regime k .

C.4 Computational Complexity

For one sweep with L combined grid points, K regimes, and P SIR particles, constructing the emission factors requires $K(L - 1)$ FNLE density evaluations, followed by $O(LK^2)$ arithmetic operations for FFBS [30]. SIR requires $O(P|\mathcal{T}_{\text{cand}}|)$ conditional FNLE samples and density evaluations across all candidate blocks. The NUTS cost depends on its trajectory-dependent number of gradient evaluations, each involving $O(L)$ local FNLE terms, together with observation and prior terms [14]. These costs exclude offline FNLE training.

D EXPERIMENTAL SETTINGS

This appendix records the settings for the synthetic-data experiments (OU comparison, LV, gene-expression CLE, and susceptible–infected–recovered) and the real-data LV analysis.

D.1 Settings for the Main Experiments

The synthetic-data LV, gene-expression CLE, and susceptible–infected–recovered experiments, along with the real-data LV analysis, use two regimes and observe every state component with Gaussian errors as in Eq. (10).

D.1.1 FNLE Training

The time-conditioned FNLE training dataset is

$$\mathcal{D}_{\text{train}} = \{(y_r^{\text{prev}}, y_r^{\text{next}}, \theta_r, \Delta_r)\}_{r=1}^{R_{\text{train}}}, \quad (54)$$

where R_{train} is the number of simulated transitions and r indexes training examples rather than positions in a trajectory. The density estimator is trained by maximizing the conditional log-likelihood

$$\max_{\phi} \sum_{r=1}^{R_{\text{train}}} \log \hat{p}_{\phi}(y_r^{\text{next}} | y_r^{\text{prev}}, \theta_r, \Delta_r). \quad (55)$$

Omitting elapsed-time conditioning gives the standard FNLE training objective.

A separate time-conditioned FNLE model was trained for each experiment. We used preliminary MCMC draws to check whether the FNLE parameter training ranges in Sections D.2.5 and D.3 covered the parameter regions relevant to inference. Where needed, we expanded the ranges, retrained FNLE, and reran MCMC. These adjustments refine the numerical likelihood approximation without changing the SSDE model or prior.

A natural cubic spline through that experiment’s observations was evaluated on an Euler–Maruyama grid [18] with step size $h = 0.01$ to obtain a reference path. Negative spline values were truncated at zero. For every training example, each component of $\log \theta$ was sampled uniformly between the logarithms of the corresponding physical-scale bounds reported below, and θ was then obtained by exponentiation. For each training example, one location was selected from the reference path. Isotropic Gaussian noise with standard deviation 0.32 was added to this state. If the perturbed state left $[0, 10^4]^D$, the noise was redrawn until the state lay inside this domain.

The maximum elapsed time required for density evaluation is bounded by the largest observation interval, which determines the required training range. The number of numerical transitions, n , was sampled uniformly from $\{1, \dots, n_{\text{max}}\}$, and the elapsed time was $\Delta = nh$. The simulator was advanced by n Euler–Maruyama steps, and FNLE was trained on the scaled increment

$$R = \frac{Y_{t+\Delta} - Y_t}{\sqrt{\Delta}} \quad (56)$$

conditional on (θ, Y_t, Δ) . When the learned density of R is evaluated as a transition density for $Y_{t+\Delta}$, the change of variables gives

$$\begin{aligned} \hat{p}_{\phi}(y^{\text{next}} | y^{\text{prev}}, \theta, \Delta) \\ = \Delta^{-D/2} \hat{p}_{\phi, R} \left(\frac{y^{\text{next}} - y^{\text{prev}}}{\sqrt{\Delta}} \middle| y^{\text{prev}}, \theta, \Delta \right). \end{aligned} \quad (57)$$

Table 3: Experiment-Specific FNLE Training Settings.

Experiment	$n_{\max}$	$n_{\max}h$
Synthetic-data OU (approx. MCMC with Euler–Maruyama FNLE)	100	1.00
Synthetic-data LV	100	1.00
Synthetic-data gene-expression CLE	25	0.25
Synthetic-data susceptible–infected–recovered	20	0.20
Real-data LV	98	0.98

Table 4: Priors for the Observation Variances

Experiment	Component	Prior
Synthetic-data OU	scalar state	$\tau^2 \sim \text{InvGamma}(10^{-3}, 10^{-3})$
Synthetic-data LV	predator	$\tau_1^2 \sim \text{InvGamma}(10^{-3}, 10^{-3})$
Synthetic-data LV	prey	$\tau_2^2 \sim \text{InvGamma}(10^{-3}, 10^{-3})$
Synthetic-data gene-expression CLE	mRNA	$\tau_1^2 \sim \text{InvGamma}(10^{-3}, 10^{-3})$
Synthetic-data gene-expression CLE	protein fluorescence	$\tau_2^2 \sim \text{InvGamma}(10^{-3}, 10^{-3})$
Synthetic-data susceptible–infected–recovered	susceptible	$\tau_1^2 \sim \text{InvGamma}(10^{-3}, 10^{-3})$
Synthetic-data susceptible–infected–recovered	recovered	$\tau_2^2 \sim \text{InvGamma}(10^{-3}, 10^{-3})$
Real-data LV	<i>Didinium</i>	$\tau_1^2 \sim \text{InvGamma}(10^{-3}, 10^{-3})$
Real-data LV	<i>Paramecium</i>	$\tau_2^2 \sim \text{InvGamma}(10^{-3}, 10^{-3})$

Thus, Eq. (56) is a target reparameterization and does not change the underlying transition model. For the OU experiment, the two FNLE training sets shared identical conditioning inputs ($y^{\text{prev}}, \theta, \Delta$) and differed only in how y^{next} was generated. For approximate MCMC with FNLE (exact-transition training), it was sampled from the exact transition density in Appendix D.2.1; for approximate MCMC with FNLE (Euler–Maruyama training), it was generated by Euler–Maruyama.

Every FNLE model was a neural spline flow [6] with 50 hidden features, five transforms, and ten spline bins. Each synthetic-data training set contained 100,000 independently sampled parameter settings, and the real-data training set contained 800,000, with one simulated transition per setting. Training used a batch size of 256, a learning rate of 5×10^{-4} , at most 100 epochs, and early stopping after 20 epochs without validation-loss improvement. The density estimator and batch size were selected through preliminary experiments, and maximum epoch counts were set according to the available computational budget. All other FNLE training hyperparameters used the `sbi` defaults, including reserving 10% of the simulated transitions for validation. The experiment-specific transition-length settings and final epoch counts are given in Table 3.

D.1.2 Prior Settings

SDE parameter priors and parameter-sharing structures are given in Sections D.2.5 and D.3. Each prior is evaluated once per unique parameter. Here, $\text{LogNormal}(m, s)$ denotes a distribution whose natural logarithm is Gaussian with mean m and standard deviation s . The initial regime probabilities were $(1/2, 1/2)$, and the initial state had an improper uniform prior, $p(y_1) \propto 1$ on $\mathbb{R}^D$.

The observation-variance priors are $\tau_d^2 \sim \text{InvGamma}(\alpha_d^\tau, \beta_d^\tau)$, with the experiment-specific settings listed in Table 4.

In all experiments, the off-diagonal CTMC rates independently followed $\text{Gamma}(2, 20)$ in the shape–rate parameterization.

D.1.3 MCMC Settings and Initialization

The deterministic uniformization rate used $c_\Omega = 3$ and a numerical safeguard $\varepsilon = 10^{-6}$ in

$$\Omega(Q) = \max \left\{ c_\Omega \max_k (-q_{kk}), \max_k (-q_{kk}) + \varepsilon, \varepsilon \right\}. \quad (58)$$

Table 5: Fixed NUTS Step Sizes by Experiment and Sampler

Experiment	State path $y_{1:L}$	Parameters Θ
Synthetic-data OU (exact MCMC)	2^{-8}	2^{-5}
Synthetic-data OU (approx. MCMC with exact FNLE)	2^{-8}	2^{-5}
Synthetic-data OU (approx. MCMC with Euler–Maruyama FNLE)	2^{-8}	2^{-5}
Synthetic-data LV	2^{-9}	2^{-8}
Synthetic-data gene-expression CLE	2^{-9}	2^{-8}
Synthetic-data susceptible–infected–recovered	2^{-9}	2^{-8}
Real-data LV	2^{-9}	2^{-5}

Table 6: MCMC Chain Counts and Lengths Used for Posterior Summaries. Sweep, burn-in, and retained-draw counts are per chain.

Experiment	Chains	Sweeps	Burn-in	Retained
Synthetic-data OU (exact MCMC)	4	10,000	5,000	5,000
Synthetic-data OU (approx. MCMC with exact FNLE)	4	10,000	5,000	5,000
Synthetic-data OU (approx. MCMC with Euler–Maruyama FNLE)	4	10,000	5,000	5,000
Synthetic-data LV	4	10,000	5,000	5,000
Synthetic-data gene-expression CLE	4	10,000	5,000	5,000
Synthetic-data susceptible–infected–recovered	4	10,000	5,000	5,000
Real-data LV	4	10,000	5,000	5,000

After computing $B = I_K + Q/\Omega(Q)$, negative entries caused by numerical error were clipped to zero and each row was renormalized. Newly inserted candidate-time states were sampled by sampling-importance-resampling with 100 particles. The continuous path and nonredundant SDE parameters were updated conditionally using Pyro NUTS. Before the production runs, we used Pyro’s step-size adaptation in preliminary runs and fixed the resulting step sizes. Table 5 lists the step sizes for the state-path and parameter updates by experiment and sampler. The production chains used one NUTS draw per sweep, with no warm-up or further step-size adaptation. In all experiments, the maximum tree depth was 5 for the state-path update and 3 for the parameter update; these values were selected in preliminary experiments based on the observed divergence rates and sample displacements. The regime path was updated by log-space forward filtering and backward sampling.

At initialization, SDE parameters were sampled from their experiment-specific priors conditional on lying inside the corresponding FNLE ranges. Shared components were sampled once and copied across regimes. Every off-diagonal CTMC rate and observation variance was sampled from its prior. The initial regime path was simulated from an auxiliary symmetric CTMC with $q_{12} = q_{21} = 0.2$ for the synthetic-data OU and LV experiments, 0.4 for the synthetic-data gene-expression CLE experiment, and 0.5 for the synthetic-data susceptible–infected–recovered and real-data LV experiments. This generator was used only to initialize the regime path, independently of the initial Q drawn from its prior. The initial continuous path was obtained by linear interpolation of the observations. FNLE training and the synthetic-data chains used random seed zero; the four real-data chains used seeds zero through three. Chain lengths are summarized in Table 6. For trajectory summaries, each sampled path was linearly interpolated between its grid points onto a common plotting grid before computing pointwise posterior credible intervals.

D.2 Synthetic-Data Experiments

D.2.1 Ornstein–Uhlenbeck Experiment

The OU experiment used Eq. (24). On an interval of length Δ with regime k fixed, define $m_k(x, u) = \mu_k + e^{-\kappa u}(x - \mu_k)$ and $v(u) = \sigma^2(1 - e^{-2\kappa u})/(2\kappa)$. The exact transition and one-time bridge distributions are

$$p(y \mid x, \theta_k, \Delta) = \mathcal{N}(y; m_k(x, \Delta), v(\Delta)), \quad (59)$$

$$Y_u \mid (Y_0 = x, Y_\Delta = y, \theta_k) \sim \mathcal{N}\left(m_k(x, u) + e^{-\kappa(\Delta-u)} \frac{v(u)}{v(\Delta)} [y - m_k(x, \Delta)], \frac{v(u)v(\Delta-u)}{v(\Delta)}\right), \quad 0 < u < \Delta. \quad (60)$$

For multiple interior times, the joint bridge distribution follows from Eq. (33) and is multivariate Gaussian. Exact MCMC uses the transition density directly, while approximate MCMC with FNLE (exact-transition training) uses it only to generate FNLE training transitions and uses the learned density at inference. The exact bridge distribution is used only by exact MCMC; both approximate samplers use the FNLE-based SIR bridge update. Conditional on this path, the state was simulated by Euler–Maruyama with step size $h = 0.01$ and independent Brownian increments.

The initial state was $Y_0 = 0.0$. The observation standard deviations were $\tau_1 = \tau_2 = 0.01$. The observation period was $T = 100$, with 101 observations at spacing 1.0.

D.2.2 Switching Lotka–Volterra Experiment

The LV experiment used Eq. (25). The regime path was simulated from a two-state CTMC with initial probabilities $(1/2, 1/2)$, exact holding times, and exact jump destinations. Conditional on this path, the state was simulated by Euler–Maruyama with step size $h = 0.01$ and independent Brownian increments. After each step, the state was restricted to $[0, 10^4]^2$ componentwise.

The initial state was $(Y_{0,1}, Y_{0,2}) = (1.0, 0.5)$, and the data-generating CTMC rates were $q_{12} = q_{21} = 0.03$. The observation standard deviations were $\tau_1 = \tau_2 = 0.01$. The observation period was $T = 100$, with 101 observations at spacing 1.0.

D.2.3 Switching Gene-Expression CLE Experiment

The gene-expression CLE experiment used Eq. (26). The regime path was simulated from a two-state CTMC with initial probabilities $(1/2, 1/2)$, exact holding times, and exact jump destinations. Conditional on this path, the state was simulated by Euler–Maruyama with step size $h = 0.01$ and independent Brownian increments. After each step, the state was restricted to $[0, 10^4]^2$ componentwise. To see the change of scale, let P_t be the continuous approximation to the protein copy number, with translation rate γ/c^2 per mRNA molecule and degradation rate δ . Its chemical Langevin equation is

$$\begin{aligned} dP_t = & \left(\frac{\gamma}{c^2} Y_{t,1} - \delta P_t \right) dt \\ & + \sqrt{\frac{\gamma}{c^2} Y_{t,1} + \delta P_t} dW_{t,2}. \end{aligned} \quad (61)$$

The transformation $Y_{t,2} = c^2 P_t$, with $c > 0$, gives Eq. (26). Thus, γ is the fluorescence-scaled production coefficient, and c^2 converts protein copy number to fluorescence intensity.

The initial state was $(Y_{0,1}, Y_{0,2}) = (100, 0.5)$, and the data-generating CTMC rates were $q_{12} = q_{21} = 0.05$. The observation standard deviations were $\tau_1 = 1.0$ and $\tau_2 = 0.01$. The observation period was $T = 50$, with 201 observations at spacing 0.25.

D.2.4 Switching Susceptible–Infected–Recovered Epidemic Experiment

The susceptible–infected–recovered experiment used Eq. (27), with fixed total population $N_{\text{pop}} = 1000$. The regime path had a single prescribed switch at the midpoint of the observation period. Conditional on this path, the state was simulated by Euler–Maruyama with step size $h = 0.01$ and independent Brownian increments. After each step, the state was restricted to $[0, 10^4]^2$ componentwise and projected onto $\{(s, r) : s \geq 0, r \geq 0, s + r \leq N_{\text{pop}}\}$.

The initial state was $(Y_{0,1}, Y_{0,2}) = (990, 0)$, implying $I_0 = 10$. The regime path was fixed to regime 1 on $[0, 20)$ and regime 2 on $[20, 40]$. The observation standard deviations were $\tau_1 = \tau_2 = 0.5$, and negative noisy observations were truncated at zero. The observation period was $T = 40$, with 201 observations at spacing 0.20. Both state components were supplied to inference and plotted. The infected population was obtained as $I_t = N_{\text{pop}} - Y_{t,1} - Y_{t,2}$.

D.2.5 Data-Generating Parameters and Inference Settings

Table 7 lists the data-generating SDE parameters, with regime-specific values ordered by regime; Table 8 gives the FNLE ranges, parameter-sharing structures, and priors.

Table 7: Data-Generating SDE Parameters for the Synthetic Experiments. Regime-specific values are listed for regimes 1 and 2.

Model	Parameter	Regime 1	Regime 2	Structure
OU	κ	0.15	0.15	shared
	μ	-1.2	1.2	regime-specific
	σ	0.05	0.05	shared
LV	α	1.0	0.5	regime-specific
	β	1.0	0.5	regime-specific
	γ	1.0	0.2	regime-specific
	δ	1.0	0.2	regime-specific
	σ_1	0.05	0.05	shared
	σ_2	0.05	0.05	shared
Gene-expression CLE	ρ	100	240	regime-specific
	β	1.0	1.0	shared
	γ	0.004	0.004	shared
	δ	0.8	0.8	shared
	c	0.03	0.03	shared
Susceptible–infected–recovered	β	0.40	0.80	regime-specific
	γ	0.10	0.20	regime-specific

For all synthetic-data experiments, regime labels were aligned to the truth by choosing the permutation that minimized the root-mean-square discrepancy between posterior mean log parameters and the data-generating log parameters.

D.3 Real-Data Experiment

The real-data analysis used 71 paired observations of *Didinium* and *Paramecium* abundance from Veilleux [34]. The observations were sorted by time, the first time was shifted to zero, and both abundance variables were divided by 100 before fitting. The observation times were irregular and spanned $[0, 35.08]$ days. The state was ordered as $Y_t = (Y_{t,1}, Y_{t,2})^\top$, where $Y_{t,1}$ and $Y_{t,2}$ denote *Didinium* and *Paramecium*, respectively.

The analysis used the LV model in Eq. (25). Table 9 gives the parameter-sharing structure, FNLE ranges, and priors.

For Figure 4, samples from chains 0–3 were pooled after aligning the regime labels.

D.3.1 Chain-Specific Posterior Distributions and Traces

Figure 5 compares the parameter distributions and traces from all four real-data chains.

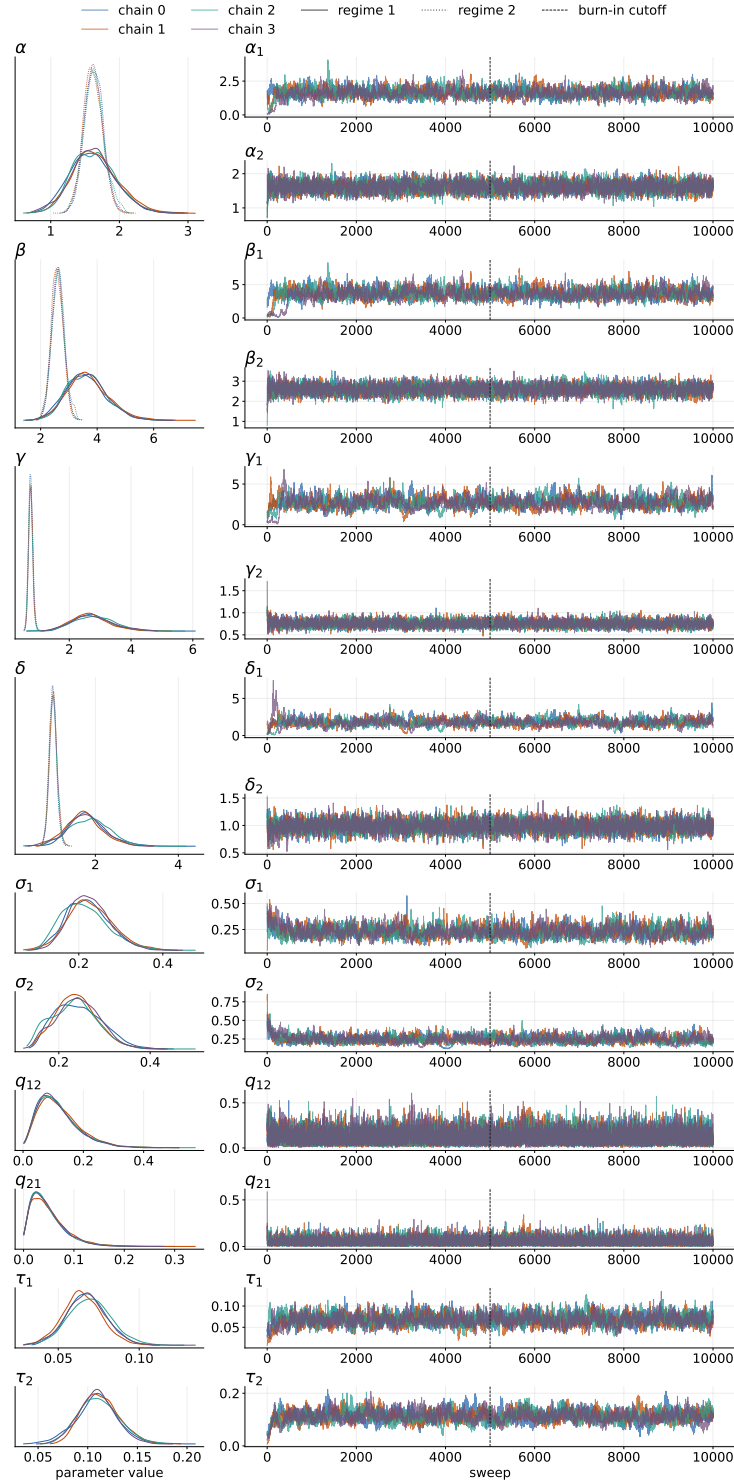


Figure 5: Multi-chain diagnostics for the real-data Lotka–Volterra analysis. Left panels show chain-specific marginal posterior density estimates, and right panels show traces over all 10,000 sweeps. Colors distinguish the four chains; solid and dotted lines distinguish regimes 1 and 2 for regime-specific parameters. Vertical dashed lines mark the burn-in cutoff at 5,000 sweeps. The observation-noise panels show standard deviations τ_1, τ_2 , not variances.

Table 8: SDE Parameter Inference Settings for the Synthetic Experiments. Parameter ranges and priors are on the physical scale.

Model	Parameter	Structure	FNLE range	Prior
OU	κ	shared	$[0.05, 0.2]$	$\text{LogNormal}(0, 1)$
	μ	regime-specific	$[-2.0, 2.0]$	$\text{Normal}(0, 1)$
	σ	shared	$[0.01, 0.1]$	$\text{LogNormal}(0, 1)$
LV	α	regime-specific	$[0.01, 2]$	$\text{LogNormal}(0, 1)$
	β	regime-specific	$[0.1, 3]$	$\text{LogNormal}(0, 1)$
	γ	regime-specific	$[0.01, 2]$	$\text{LogNormal}(0, 1)$
	δ	regime-specific	$[0.1, 3]$	$\text{LogNormal}(0, 1)$
	σ_1	shared	$[0.01, 0.30]$	$\text{LogNormal}(-1, 1)$
	σ_2	shared	$[0.01, 0.20]$	$\text{LogNormal}(-1, 1)$
Gene-expression CLE	ρ	regime-specific	$[50, 400]$	$\text{LogNormal}(4, 1)$
	β	shared	$[0.5, 1.5]$	$\text{LogNormal}(0, 1)$
	γ	shared	$[0.001, 0.010]$	$\text{LogNormal}(0, 1)$
	δ	shared	$[0.5, 1.5]$	$\text{LogNormal}(0, 1)$
	c	shared	$[0.01, 0.10]$	$\text{LogNormal}(0, 1)$
Susceptible–infected–recovered	β	regime-specific	$[0.05, 2]$	$\text{LogNormal}(0, 1)$
	γ	regime-specific	$[0.02, 2]$	$\text{LogNormal}(0, 1)$

Table 9: SDE Parameter Settings for the Real-Data LV Analysis. Parameter ranges and priors are on the physical scale.

Parameter	Structure	FNLE range	Prior
α	regime-specific	$[0.1, 4]$	$\text{LogNormal}(0, 1)$
β	regime-specific	$[0.01, 6]$	$\text{LogNormal}(0, 1)$
γ	regime-specific	$[0.1, 8]$	$\text{LogNormal}(0, 1)$
δ	regime-specific	$[0.01, 5]$	$\text{LogNormal}(0, 1)$
σ_1	shared	$[0.01, 0.8]$	$\text{LogNormal}(-1, 1)$
σ_2	shared	$[0.01, 2]$	$\text{LogNormal}(-1, 1)$

D.4 MCMC Diagnostics

Table 10: Per-parameter MCMC diagnostics for the three OU samplers on synthetic data.

OU sampler	Parameter	$\hat{R}$	Bulk ESS	Tail ESS
Exact MCMC	κ	1.015	198.4	1,004.2
	μ_1	1.004	1,187.8	6,199.0
	μ_2	1.005	1,478.3	5,050.6
	σ	1.004	1,519.8	2,234.7
	q_{12}	1.000	16,311.0	19,035.3
	q_{21}	1.000	16,972.1	18,373.4
	τ	1.007	787.0	1,386.9
Approx. MCMC with FNLE (exact-transition training)	κ	1.011	323.9	2,294.4
	μ_1	1.002	5,909.5	13,911.3
	μ_2	1.006	792.1	3,312.8
	σ	1.004	778.2	1,338.5
	q_{12}	1.001	11,410.3	16,523.3
	q_{21}	1.001	12,109.2	19,161.3
	τ	1.017	405.4	1,032.6
Approx. MCMC with FNLE (Euler–Maruyama training)	κ	1.021	229.2	1,903.9
	μ_1	1.006	1,497.8	4,256.4
	μ_2	1.004	2,159.5	7,742.9
	σ	1.004	953.9	1,984.0
	q_{12}	1.000	14,125.3	16,715.3
	q_{21}	1.000	12,237.1	19,034.8
	τ	1.008	487.4	838.5

Table 11: Per-parameter MCMC diagnostics for the synthetic Lotka–Volterra experiment.

Parameter	$\hat{R}$	Bulk ESS	Tail ESS
α_1	1.006	421.2	1,090.5
α_2	1.007	406.2	777.6
β_1	1.006	395.9	973.0
β_2	1.006	385.9	771.1
γ_1	1.028	217.1	893.4
γ_2	1.038	143.4	301.5
δ_1	1.037	128.9	726.6
δ_2	1.048	107.1	318.3
σ_1	1.046	66.8	105.1
σ_2	1.056	65.5	45.1
q_{12}	1.004	2,537.5	12,637.6
q_{21}	1.006	1,251.7	5,873.1
τ_1	1.016	316.6	1,082.3
τ_2	1.016	400.5	742.1

Table 12: Per-parameter MCMC diagnostics for the gene-expression CLE experiment.

Parameter	$\hat{R}$	Bulk ESS	Tail ESS
ρ_1	1.003	805.9	1,595.1
ρ_2	1.001	1,274.5	3,142.3
β	1.011	211.0	333.4
γ	1.007	334.2	691.6
δ	1.007	335.3	681.7
c	1.018	201.7	358.9
q_{12}	1.000	16,712.8	18,545.4
q_{21}	1.000	18,588.0	19,235.4
τ_1	1.697	6.2	11.5
τ_2	1.003	725.2	2,003.8

Table 13: Per-parameter MCMC diagnostics for the susceptible–infected–recovered experiment.

Parameter	$\hat{R}$	Bulk ESS	Tail ESS
β_1	1.019	481.9	1,013.8
β_2	1.054	61.6	338.2
γ_1	1.065	66.2	485.9
γ_2	1.003	645.6	1,278.9
q_{12}	1.000	15,315.1	18,870.5
q_{21}	1.001	10,209.2	14,825.5
τ_1	1.539	7.0	41.1
τ_2	1.298	10.8	38.8

Table 14: Per-parameter MCMC diagnostics for the real-data Lotka–Volterra analysis.

Parameter	$\hat{R}$	Bulk ESS	Tail ESS
α_1	1.004	442.4	987.0
α_2	1.005	929.8	909.5
β_1	1.003	466.3	1,065.3
β_2	1.005	871.0	888.9
γ_1	1.011	298.2	732.9
γ_2	1.007	1,160.2	2,167.1
δ_1	1.014	296.4	493.5
δ_2	1.004	1,563.6	3,540.1
σ_1	1.027	342.6	478.5
σ_2	1.010	276.9	552.2
q_{12}	1.002	5,023.5	9,473.7
q_{21}	1.002	3,481.9	7,393.4
τ_1	1.014	397.2	683.3
τ_2	1.010	183.5	344.5

D.5 Computing Environment and Software

D.5.1 Computing Infrastructure

All computations used only the CPU on a local PC with an Intel Core i7-1195G7 processor (4 cores, 8 threads) and 32GiB of RAM, running Ubuntu 22.04.5 LTS.

D.5.2 Software and Existing Assets

We train the local transition densities for FNLE using the NLE implementation in `sbi` (v0.24.0) [4]. We additionally use the `Pyro` package (v1.9.1) [1] for posterior sampling with NUTS. We compute $\hat{R}$ and bulk and tail ESS using `ArviZ` (v0.22.0) [21]. All three packages are distributed under the Apache License 2.0. Complete software dependencies and their exact versions are provided in the supplementary material via the accompanying `uv.lock` file. The real data used in this study are publicly available in the `rEDM` R package (BSD 2-Clause license), originally from Veilleux [34].